

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra aplikované matematiky

**Implementace Mortarových prvků a
metody Internodes pro nesouhlasné
sítě do programového balíku
PERMON**

**Implementation of Mortar elements
and the Internodes method for
dissimilar meshes in the PERMON
software package**

Zadání diplomové práce

Student:

Bc. Matouš Blažek

Studijní program:

N2647 Informační a komunikační technologie

Studijní obor:

1103T031 Výpočetní matematika

Téma:

Implementace Mortarových prvků a metody Internodes pro nesouhlasné sítě do programového balíku PERMON
Implementation of Mortar elements and the Internodes method for dissimilar meshes in the PERMON software package

Jazyk vypracování:

čeština

Zásady pro vypracování:

V dnešní době se snažíme řešit stále více reálných úloh. Dokážeme řešit úlohy s miliardou neznámých na superpočítačích, ale tyto úlohy jsou většinou modelového typu, kde se dopouštíme řady zjednodušení. Cestou, jak některá zjednodušení eliminovat, je použití nesouhlasných sítí a to jak pro lepení podoblastí, tak na podmínky nepronikání, které se objevují u kontaktních úloh. Jedna z možností jak s těmito nesouhlasnými sítěmi pracovat je použití mortarů nebo metody internode.

Cílem práce je nastudování problematiky mortarů a internodes pro úlohy ve 2D a následná implementace do programového balíku PERMON, který je vyvíjený na Katedře aplikované matematiky a Ústavu geoniky, Akademie věd České republiky. Student se seznámí s matematickým aparátem, který stojí za problematikou mortarů a internodes a své nabyté znalosti zúročí při implementaci. Následně svou implementaci podrobí testování na sadě vybraných reálných úloh.

Seznam doporučené odborné literatury:

Ben Belgacem F., The mortar finite element method with Lagrange multipliers, Numer. Math., 84 (2) (1999), pp. 173-197.

Wohlmuth B.I., A mortar finite element method using dual spaces for the Lagrange multiplier, SIAM J. Numer. Anal., 38 (2000), pp. 989-1012.

Deparis S., Forti D., Gervasio P., Quarteroni A., INTERNODES: an accurate interpolation-based method for coupling the Galerkin solutions of PDEs on subdomains featuring non-conforming interfaces, Comput. & Fluids, 141 (2016), pp. 22-41.

Gervasio P., Quarteroni A., Analysis of the INTERNODES method for non-conforming discretizations of elliptic equations, Computer Methods in Applied Mechanics and Engineering, 334 (2018), pp. 138-166.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **doc. Ing. Martin Čermák, Ph.D.**

Datum zadání: 01.09.2019

Datum odevzdání: 30.04.2020



prof. RNDr. Jiří Bouchala, Ph.D.
vedoucí katedry

prof. Ing. Pavel Brandštetter, CSc.
děkan fakulty

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně. Uvedl jsem všechny literární
prameny a publikace, ze kterých jsem čerpal.

V Ostravě 15. května 2020


.....

Rád bych zde poděkoval své rodině za umožnění studia na vysoké škole a podporu. Chtěl bych také poděkovat vedoucímu mé diplomové práce panu doc. Ing. Martinu Čermákovi, Ph.D. za klidný průběh, motivaci, pomoc a čas, který mi věnoval po celou dobu vedení této práce.

Abstrakt

Tato diplomová práce je zaměřena na Mortar metodu a metodu Internodes pro lineární eliptické okrajové úlohy ve 2D. Tyto metody se používají pro lepení nekonformních diskretizačních sítí v metodách rozložení oblastí. Pochopením a praktickou implementací těchto metod můžeme rozšířit funkcionalitu stávajících metod, které používají „pěkné“ konformní sítě. Obě metody jsou zde popsány teoreticky i prakticky. Nakonec porovnáme a zhodnotíme klíčové vlastnosti obou metod.

Klíčová slova: Mortar metoda, Internodes, nesouhlasné sítě, nekonformní sítě, metody rozložení oblasti

Abstract

This diploma thesis is focused on Mortar method and Internodes for linear elliptic boundary problems in 2D. These methods are used for nonconforming domain coupling in domain decomposition methods. After learning and implementing these methods, we can extend functionality of existing methods for nonconforming cases. We will also review and compare each of these methods at the end of this thesis.

Keywords: Mortar element method, Internodes, non-matching grids, non-conforming discretization, domain decomposition methods

Obsah

Seznam použitých zkratk a symbolů	8
Seznam obrázků	9
Seznam výpisů zdrojového kódu	10
1 Úvod	11
2 Formulace úlohy	13
2.1 Variační formulace	14
2.2 Sedlobodová úloha	16
2.3 Společná část	19
3 Mortar metoda konečných prvků	22
3.1 Popis metody	22
3.2 Algebraická formulace mortar podmínky	25
3.3 Implementace	26
4 Metoda Internodes	33
4.1 Popis metody	33
4.2 Interpolace	34
4.3 Formulace	36
4.4 Algebraická formulace	37
4.5 Implementace	39
5 Numerické experimenty	41
5.1 Membrána	41
5.2 Lineární elasticita	43
6 Závěr	45
Literatura	46
Přílohy	48

Seznam použitých zkratek a symbolů

I	– jednotková matice (identita)
O	– nulová matice
\mathbf{o}	– nulový vektor
A^T	– transponovaná matice A
A^{-1}	– inverzní matice A
(\mathbf{u}, \mathbf{v})	– skalární součin vektorů \mathbf{u}, \mathbf{v}
$ x $	– absolutní hodnota čísla x
$ \mathbf{v} $	– velikost vektoru \mathbf{v}
\mathbb{R}	– množina reálných čísel
Ω	– neprázdná omezená oblast
$\partial\Omega$	– hranice oblasti
\bar{X}	– uzávěr množiny
$X \setminus Y$	– rozdíl množin
RBF	– radiální bázová funkce

Seznam obrázků

1	Příklad geometricky konformního a nekonformního dělení	19
2	Příklad souhlasných a nesouhlasných sítí	20
3	Příklady nekonformity	21
4	Příklad geometricky konformního dělení hran	22
5	Nekonformní příklad rozdělení hran	23
6	Příklad hrany Γ_k	24
7	Vizualizace bázových funkcí Ψ^h	24
8	Segment	26
9	Tradiční segmentace	27
10	Nedokonalost tradiční segmentace	27
11	Normálový směr	28
12	Výsledná robustní segmentace	29
13	Robustní segmentace	29
14	Radiální bázová funkce	36
15	Interpolace uzlů sítí pomocí RB	36
16	Experiment 1	41
17	Experiment 2: metoda Mortarů pro různé nastavení kombinace master-slave hran	42
18	Experiment 2: metoda Internodes pro různé nastavení kombinace master-slave hran	42
19	Experiment 3	43
20	Experiment 4: metoda Mortarů	44
21	Experiment 4: metoda Internodes	44

Seznam výpisů zdrojového kódu

1 Úvod

Prvním krokem řešení každého problému matematické fyziky je vytvořit adekvátní matematický model. Ten může být více či méně dokonalým popisem daného jevu. Obvykle se začíná s jednodušším modelem, který některé jevy zanedbává. Poté co se tento jednoduchý model naučíme řešit postupujeme k modelům, které jsou schopny zachytit složitější jevy. Typickým příkladem v mechanice kontinua tuhé fáze je Lamého systém lineární pružnosti, který dobře charakterizuje chování některých materiálů za určitých podmínek.

Matematici a inženýři zkoumají jednotlivé modely z různých pohledů. Matematici se na model dívají z pohledu kvalitativního – řešitelnost, jednoznačnost řešení a jeho vlastnosti, např. spojitá závislost řešení na vstupních parametrech. Významným přínosem matematiků v této oblasti je využití prostředků konvexní analýzy pro formulaci úloh a jejich následné numerické řešení. Na druhé straně inženýři se na model dívají z pohledu spíše kvantitativního neboli analýzu numerických výsledků pro konkrétní data a tím verifikaci a platnost daného modelu.

Velkým přínosem pro řešení jednotlivých modelů se stala metoda konečných prvků [18], která umožnila řešit stále složitější a rozsáhlejší úlohy, a to nejen v lineární elasticitě. V dnešní době se setkáváme s úlohami, které mají miliony až miliardy stupňů volnosti a bez paralelního zpracování a následných paralelních výpočtů se u těchto úloh jen těžko obejdeme. Existuje řada způsobů a metod, jak dané problémy řešit paralelně. Jeden z nich je použití metody na rozdělení na podoblasti. Můžeme si vybrat mezi překrývající metodou (Schwarzova metoda) nebo nepřekrývající metodou (metody typu FETI [12]). V této práci jsme se zaměřili na nepřekrývající metody a to převážně z důvodu, že na Katedře aplikované matematiky se již téměř dvě desetiletí touto metodou zabývají a také se členové katedry podílejí na open-sourcovém programovém balíku PERMON (<http://permon.vsb.cz>) [16, 17, 26], který využívá metodu Total-FETI [9] pro miliardy stupňů volnosti.

PERMON je softwarový balík, který se soustředí na implementaci optimálních řešičů úloh kvadratického programování pro masivně paralelní výpočty. Klíčovou úlohou pro testování algoritmů je numerické řešení kontaktních úloh lineární elasticity. Nicméně aktuální stav implementace obsahuje pouze úlohy diskretizované na souhlasných sítích, což je velice omezující. Zejména proto, že v případě souhlasných sítí mají jednotlivé objekty v odpovídající optimalizační úloze příliš „pěkný“ tvar. Řešení úloh na nekonformních sítích tedy představuje úlohy, kde numerické chování paralelních algoritmů ještě nebylo podrobena testování. Nicméně pouze pochopení celkové problematiky sestavování objektů v nové odpovídající úloze může vést k odvození klíčových vlastností úlohy a následnou diskuzi nad získanými výsledky. Možností, jak tyto omezení odstranit je použití metody Mortarů, nebo metody Internodes, které umí řešit úlohy na nesouhlasných sítích a umožní rozšířit funkcionalitu softwaru PERMON na další úroveň.

Původně se předpokládalo, že se dané metody naprogramují a vyzkoušejí v prostředí MATLAB pro membrány, a pak se naprogramují do PERMONu. S vedoucím práce jsme se ale dohodli, že implementace do PERMONu se zatím dělat nebude a raději se vyzkouší funkčnost těchto metod

také pro úlohy lineární elasticity, kterou PERMON umí také řešit. Výsledná implementace do PERMONu se provede pro membrány i lineární elasticitu po odevzdání a obhájení magisterské práce.

2 Formulace úlohy

Na začátek připomeneme několik pojmů z funkcionální analýzy, abychom problematice Mortarů, Internodes a variační formulaci lépe porozuměli. Následující pojmy jsou převzaty z [19].

Definice 1 (Prostor L^p) *Nechť (X, \mathcal{A}, μ) je prostor s mírou a $1 \leq p < \infty$. Prostor $L^p(X)$ obsahuje všechny třídy ekvivalentních měřitelných funkcí $f : X \rightarrow \mathbb{R}$ takových, že*

$$\int_X |f|^p d\mu < \infty,$$

kde dvě měřitelné funkce jsou ekvivalentní, pokud se rovnají skoro všude na X . Příslušná norma je definována takto

$$\|u\|_p = \left(\int_X |f|^p d\mu \right)^{1/p}.$$

Definice 2 (Sobolevův prostor) *Nechť $\Omega \subset \mathbb{R}^d$, $d \in \mathbb{N}$ je neprázdná oblast. Dále nechť $k \in \{0\} \cup \mathbb{N}$ a $1 \leq p < \infty$. Poté Sobolevův prostor $W^{k,p}(\Omega)$ definujeme jako zúplnění prostoru $C^\infty(\overline{\Omega})$ vzhledem k normě*

$$\|u\|_{k,p} = \left(\sum_{|\alpha| \leq k} \int_\Omega |D^\alpha u(x)|^p dx \right)^{1/p},$$

kde α je tzv. multiindex a symbol D^α označuje zobecněnou derivaci podle α .

Poznámka 1 Pod označením H^k budeme rozumět Sobolevův prostor $W^{k,2}$.

Konkrétně nás bude zajímat Hilbertův prostor

$$H^1(\Omega) = W^{1,2}(\Omega) = \left\{ u \in L^2(\Omega) : \frac{\partial u}{\partial x_i} \in L^2(\Omega), i = 1, 2, \dots, n \right\},$$

s příslušným skalárním součinem

$$(u, v)_1 = \int_\Omega u(x)v(x) dx + \int_\Omega \nabla u(x) \cdot \nabla v(x) dx$$

a indukovanou normou

$$\|u\|_{1,2} = \left(\int_\Omega u(x)^2 d\Omega + \int_\Omega \left(\sum_{i=1}^n \left(\frac{\partial u}{\partial x_i}(x) \right)^2 \right) dx \right)^{1/2}.$$

Abychom mohli mluvit o hodnotách funkce na hranici oblasti (označujeme $\partial\Omega$), potřebujeme zavést pojem stopy funkce.

Věta 1 (O stopách) *Nechť $\Omega \subset \mathbb{R}^d$ je neprázdná omezená oblast s lipschitzovskou hranicí. Pak existuje právě jedno spojité lineární zobrazení $T : W^{1,p}(\Omega) \rightarrow L^p(\partial\Omega)$, tak, že*

$$\forall u \in C^\infty(\bar{\Omega}) : Tu = u|_{\partial\Omega}.$$

Prvek $Tu \in L^p$ nazýváme stopou funkce $u \in W^{1,p}(\Omega)$ [19]. Pokud pro dvě funkce $u, v \in W^{1,p}(\Omega)$ platí $Tu = Tv$ řekneme, že u a v se rovnají ve smyslu stop. Rovnost ve smyslu stop budeme zapisovat $u = v$ na $\partial\Omega$. Pokud má daná oblast Ω lipschitzovskou hranici, pak tvoří stopy funkcí $u \in H^1(\Omega)$ prostor stop $H^{1/2}(\partial\Omega)$, který je navíc Hilbertův prostor hustý v $L^2(\partial\Omega)$ [3].

Definice 3 (prostor stop) *Nechť $\Omega \subset \mathbb{R}^d$ je neprázdná omezená oblast s lipschitzovskou hranicí. Prostor $H^{1/2}(\partial\Omega) = \{u = Tv : v \in H^1(\Omega)\}$ nazveme prostorem stop prostoru $H^1(\Omega)$.*

Dále v tomto textu budeme uvažovat prostor $H^{-1/2}(\partial\Omega)$, který budeme chápat jako duální prostor k prostoru stop $H^{1/2}(\partial\Omega)$.

2.1 Variační formulace

Uvedeme velmi stručně variační formulaci okrajové úlohy. Budeme zde uvažovat lineární eliptickou okrajovou úlohu. Speciálními případy této úlohy jsou Laplaceova a Poissonova úloha. Laplaceova úloha slouží například k modelování vedení tepla, nebo průhybu membrány.

Nechť $\emptyset \neq \Omega \subset \mathbb{R}^2$ je omezená oblast s lipschitzovskou hranicí $\partial\Omega$. Části hranice, kde budeme vyžadovat splnění okrajové podmínky označíme jako $\partial\Omega_D$ pro část s Dirichletovou okrajovou podmínkou a $\partial\Omega_N$ pro část s Neumannovou okrajovou podmínkou. Pro celou hranici oblasti platí $\partial\Omega = \partial\Omega_D \cup \partial\Omega_N$, kde $\partial\Omega_D$ a $\partial\Omega_N$ jsou disjunktní části hranice. Jsou dány funkce $f \in L^2(\Omega)$, $p, q \in L^\infty(\Omega)$, $g_D \in H^{1/2}(\partial\Omega_D)$ a $g_N \in H^{-1/2}(\partial\Omega_N)$.

$$\begin{aligned} -\nabla \cdot (p(x)\nabla u) - q(x)u &= f(x) \quad \text{na } \Omega \\ u &= g_D(x) \quad \text{na } \partial\Omega_D, \quad p(x)\frac{\partial u}{\partial \mathbf{n}_\Omega} = g_N(x) \quad \text{na } \partial\Omega_N, \end{aligned} \tag{1}$$

kde \mathbf{n}_Ω je jednotkový vektor vnější normály hranice oblasti $\partial\Omega$. Definujeme následující prostory funkcí

$$V = \{u \in H^1(\Omega) : u = 0 \text{ na } \partial\Omega_D\}, \quad V_D = \{u \in H^1(\Omega) : u = g_D \text{ na } \partial\Omega_D\}.$$

Funkce u je slabým řešením úlohy (1) platí-li, že $u \in V_D$ a

$$a(u, v) = f(v) \quad \forall v \in V, \tag{2}$$

kde

$$a(u, v) = \int_{\Omega} (p(x)\nabla u \nabla v + q(x)uv) dx$$

$$f(v) = \int_{\Omega} f v dx + \int_{\partial\Omega_N} g_N(x)Tv ds$$

Zobrazení $a(u, v) : V \times V \rightarrow \mathbb{R}$ je bilineární forma a $f(v) : V \rightarrow \mathbb{R}$ je lineární funkcionál. Pokud je $a(u, v)$ symetrická forma, můžeme slabé řešení ekvivalentně formulovat jako minimalizační úlohu (např. pomocí Ritzovy metody). Uvažujme energetický (kvadratický) funkcionál $J(v) = \frac{1}{2}a(v, v) - f(v)$. Pro slabé řešení u platí, že $u \in V_D$ a

$$u = \min_{v \in V} J(v). \quad (3)$$

Obě formulace (2), (3) jsou za uvedených předpokladů ekvivalentní [2]. Za předpokladu, že bilineární forma $a(u, v)$ je spojitá a koercivní na V a $f(v)$ je spojitý lineární funkcionál existuje právě jedno slabé řešení splňující (2). Existence a jednoznačnost slabého řešení plyne z Lax-milgramova lemmatu. Více o existenci a jednoznačnosti slabého řešení pojednává například [19].

Dále se budeme věnovat případu, kdy máme oblast Ω rozloženou na dvě podoblasti. Postup bude podobný jako pro případ jedné oblasti. Pro jednoduchost budeme uvažovat Poissonův problém s homogenní Dirichletovou okrajovou podmínkou, což je speciálním případem výše popsané okrajové úlohy.

$$\begin{aligned} -\Delta u &= f & \text{na } \Omega \\ u &= 0 & \text{na } \partial\Omega \end{aligned} \quad (4)$$

Rozdělme oblast Ω na dvě neprázdné podoblasti Ω_1 a Ω_2 tak, že $\bar{\Omega} = \bar{\Omega}_1 \cup \bar{\Omega}_2$, $\Omega_1 \cap \Omega_2 = \emptyset$. Poznamenejme, že toto dělení je geometricky konformní a platí $\Gamma_{1,2} = \Gamma_{2,1} = \Gamma$. Pro řešení na celé oblasti Ω je třeba přidat podmínky, které vynutí hladkost řešení na rozhraní $\Gamma = \bar{\Omega}_1 \cap \bar{\Omega}_2$. Úloha pro dvě podoblasti má tento tvar:

$$-\Delta u_k = f_k \quad \text{na } \Omega_k, \quad k = \{1, 2\} \quad (5a)$$

$$u = 0 \quad \text{na } \partial\Omega \quad (5b)$$

$$u_1 - u_2 = 0 \quad \text{na } \Gamma \quad (5c)$$

$$\frac{\partial u_1}{\partial \mathbf{n}_1} + \frac{\partial u_2}{\partial \mathbf{n}_2} = 0 \quad \text{na } \Gamma \quad (5d)$$

Označení u_k chápeme jako restrikcí funkce u na Ω_k a \mathbf{n}_k jako jednotkový vektor vnější normály k oblasti Ω_k . Podmínky (5c), (5d) zaručí, že celkové řešení u včetně derivací $\frac{\partial u}{\partial \mathbf{n}}$ bude na rozhraní spojitě. Budeme pokračovat slabou formulací této úlohy. Řešení hledáme v prostoru

$$V_0 = \{u \in L^2(\Omega) : u|_{\Omega_k} \in H^1(\Omega_k), u = 0 \text{ na } \partial\Omega\}.$$

Z pohledu variačních metod stačí splnit pouze jednu z podmínek na Γ . Budeme-li předpokládat, že $u \in V_0$ a splňuje (5a), (5b), (5c), potom $u \in H_0^1(\Omega)$ a podmínka (5d) je splněna automaticky, protože z (5a) plyne $\nabla u \in H^1(\text{div}; \Omega)$ [10].

Poznámka 2 Tento Sobolevův prostor se velmi často používá při analýze smíšených metod. $H^1(\text{div}; \Omega) = \{u \in H^1(\Omega) : \text{div } u \in H^1(\Omega)\}$

Podmínku (5c) začleníme do řešení pomocí Lagrangeových multiplikátorů. Slabá formulace úlohy (5a), (5b) včetně rovnostní vazby je potom následující. Najít $u \in V_0$ a $\lambda \in H^{-1/2}(\Gamma)$ tak, že

$$\begin{aligned} a(u, v) + b(v, \lambda) &= f(v), \quad \forall v \in V_0 \\ b(v, \mu) &= 0, \quad \forall \mu \in H^{-1/2}(\Gamma), \end{aligned} \quad (6)$$

kde

$$a(u, v) = \sum_{k=1}^2 \int_{\Omega_k} \nabla u_k \nabla v_k \, dx \quad (7a)$$

$$b(v, \lambda) = \int_{\Gamma} (v_1 - v_2) \lambda \, ds \quad (7b)$$

$$b(u, \mu) = \int_{\Gamma} (u_1 - u_2) \mu \, ds \quad (7c)$$

$$f(v) = \sum_{k=1}^2 \int_{\Omega_i} f_k v_k \, dx \quad (7d)$$

Nyní se podrobněji podíváme na jednotlivé části (6). Integrál (7a) vyjadřuje slabou formu rovnice (5a) přes všechny podoblasti Ω_k . Výrazy (7b), (7c) mají oba význam rovnostní podmínky (5c). Z definice (3) rozdíl $u_1 - u_2$ na rozhraní Γ patří do $H^{1/2}(\Gamma)$. Lagrangeovy multiplikátory λ a μ potom patří do prostoru $H^{-1/2}(\Gamma)$. Otázka existence, jednoznačnosti a odhadu chyby řešení sedlobodové (smíšené) úlohy (6) souvisí s Ladyzhenskaya–Babuška–Brezzi podmínkou, která je popsána včetně důkazu v publikaci [11].

2.2 Sedlobodová úloha

Sedlobodová úloha je velmi oblíbená v inženýrské praxi a má řadu výhod, které si postupně popíšeme. Úloha je definována pro funkcionál $L(u, v)$ dvou argumentů. Cílem je najít dvojici (řešení) $(\hat{x}, \hat{\lambda})$ tak, aby hodnota funkcionálu L byla vůči prvnímu argumentu minimalizována a vůči druhému maximalizována, neboli

$$\begin{aligned} L(\hat{x}, \lambda) &\leq L(\hat{x}, \hat{\lambda}) \leq L(x, \hat{\lambda}) \quad \forall x, \lambda \\ L(\hat{x}, \hat{\lambda}) &= \sup_{\lambda} \inf_x L(x, \lambda). \end{aligned} \quad (8)$$

Tento typ úloh se vyskytuje u metod rozložení oblasti, kde metodu konečných prvků chápeme jako minimalizační úlohu a zároveň požadujeme splnění vazebních podmínek pomocí Lagran-

geových multiplikátorů. Mějme tedy minimalizační úlohu

$$\min_{\mathbf{x} \in \Omega} \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} - \mathbf{b}^T \mathbf{x}, \quad \Omega = \{\mathbf{x} \in \mathbb{R}^{n_1} : \mathbf{B} \mathbf{x} = \mathbf{c}\}, \quad (9)$$

kde $\mathbf{A} \in \mathbb{R}^{n_1 \times n_1}$ je matice tuhosti (obecně matice soustavy, kterou chceme řešit), $\mathbf{b} \in \mathbb{R}^{n_1}$ je vektor pravé strany, $\mathbf{B} \in \mathbb{R}^{n_2 \times n_1}$ ($n_1 > n_2$) je vazební matice a $\mathbf{c} \in \mathbb{R}^{n_2}$ je vektor vazebních podmínek. Matice \mathbf{B} a vektor \mathbf{c} obsahuje Dirichletovy podmínky na hranici oblasti a podmínky spojitosti na rozhraní podoblastí. Tyto rovnostní podmínky jsou vynuceny Lagrangeovými multiplikátory. Příslušný Lagrangeián a Karush-Kuhn-Tuckerova podmínka této úlohy je

$$\begin{aligned} L(\mathbf{x}, \boldsymbol{\lambda}) &= \frac{1}{2} \mathbf{x}^T \mathbf{A} \mathbf{x} - \mathbf{b}^T \mathbf{x} + \boldsymbol{\lambda}^T (\mathbf{B} \mathbf{x} - \mathbf{c}) \\ \nabla_{\mathbf{x}} L(\mathbf{x}, \boldsymbol{\lambda}) &= \mathbf{A} \mathbf{x} - \mathbf{b} + \mathbf{B}^T \boldsymbol{\lambda} = \mathbf{o} \\ \nabla_{\boldsymbol{\lambda}} L(\mathbf{x}, \boldsymbol{\lambda}) &= \mathbf{B} \mathbf{x} - \mathbf{c} = \mathbf{o}. \end{aligned} \quad (10)$$

Pro podrobnější informace k (10) odkazujeme na text [23]. Úloha (9) je ekvivalentní se sedlobodovou úlohou (8) [20] a zároveň ekvivalentní s problémem najít $(\hat{\mathbf{x}}, \hat{\boldsymbol{\lambda}}) \in \mathbb{R}^{n_1} \times \mathbb{R}^{n_2}$ splňující

$$\begin{bmatrix} \mathbf{A} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{c} \end{bmatrix}. \quad (11)$$

Nutná a postačující podmínka řešitelnosti soustavy (11) je následující [20]

$$\begin{aligned} \ker \mathbf{B}^T &= \mathbf{o}, \\ \ker \mathbf{A} \cap \ker \mathbf{B} &= \mathbf{o}. \end{aligned} \quad (12)$$

Primární formulaci (11) lze převést na duální formulaci, která je často lépe podmíněná a má menší dimenzi, za cenu složitějších matic. Duální formulace je základem FETI metod [23]. Nyní blíže popíšeme matice v soustavě (11). Matice \mathbf{B} obsahuje podmínky spojitosti na $\Gamma_{i,j}$ ve slabém smyslu

$$\int_{\Gamma_{i,j}} (u_i - u_j) q_{ij} ds = 0 \quad \forall q_{ij} \in L^2(\Gamma_{i,j}) \quad (13)$$

a Dirichletovu okrajovou podmínku na $\partial\Omega$

$$\int_{\partial\Omega \cap \partial\Omega_i} u_i q_i ds = 0 \quad \forall q_i \in L^2(\partial\Omega \cap \partial\Omega_i)$$

[21]. Velice výstižná publikace shrnující fakta o sedlobodové (smíšené) formulaci je [11].

Ted se zamyslíme nad podobou sedlobodové úlohy pro obecný počet podoblastí. Rovnici (11) rozepíšeme takto

$$\begin{bmatrix} \mathbf{K} & \mathbf{B}^T \\ \mathbf{B} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u} \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f} \\ \mathbf{c} \end{bmatrix} \iff \begin{bmatrix} \mathbf{A}_1 & \mathbf{O} & \mathbf{O} & \cdots & \mathbf{B}_1^T \\ \mathbf{O} & \mathbf{A}_2 & \mathbf{O} & \cdots & \mathbf{B}_2^T \\ \mathbf{O} & \mathbf{O} & \mathbf{A}_3 & \cdots & \mathbf{B}_3^T \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{B}_3 & \cdots & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_3 \\ \vdots \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_3 \\ \vdots \\ \mathbf{c} \end{bmatrix} \quad (14)$$

Maticе \mathbf{A}_i představuje matici tuhosti podoblasti Ω_i , matice \mathbf{B}_i jsou matice vazebních a Dirichletových okrajových podmínek související s podoblastí Ω_i . Blokově diagonální matice obsahující matice tuhosti jednotlivých podoblastí se v literatuře označuje \mathbf{K} . Maticí \mathbf{B} bez indexu budeme značit všechny vazební matice \mathbf{B}_i .

Nyní uvedeme konkrétní příklad. Budeme uvažovat případ pro dvě podoblasti Ω_1, Ω_2 . Předpokládejme jednotné číslování uzlů diskretizace dekomponované oblasti. Soustava lineárních rovnic představující tento případ je

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{O} & \mathbf{B}_1^T \\ \mathbf{O} & \mathbf{A}_2 & \mathbf{B}_2^T \\ \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \lambda \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{c} \end{bmatrix}. \quad (15)$$

Jak jsme již popsali výše matice \mathbf{A}_i jsou matice tuhosti. Protože pracujeme jen s dvěma podoblastmi nemůže dojít ke geometrické nekonformitě (více o nekonformitě v následující podkapitole). Navíc budeme předpokládat souhlasné rozhraní mezi podoblastmi. Existuje mnoho způsobů jak svázat uzly na rozhraní podoblastí. Ukážeme jeden z nejjednodušších typů vazeb, které můžeme přímo zakomponovat do matic \mathbf{B}_i . Konkrétně vyjdeme z rovnostní podmínky (5c). Podmínku budeme interpretovat na úrovni uzlů na rozhraní podoblastí. Označme vektory neznámých \mathbf{u}_1^Γ a \mathbf{u}_2^Γ . Pořadí prvků vektorů v obou vektorech je takové, aby na pozici i byly uzly, které si geometricky odpovídají. Díky souhlasným sítím můžeme přímo napsat rovnostní vazbu

$$\mathbf{u}_1^\Gamma = \mathbf{u}_2^\Gamma. \quad (16)$$

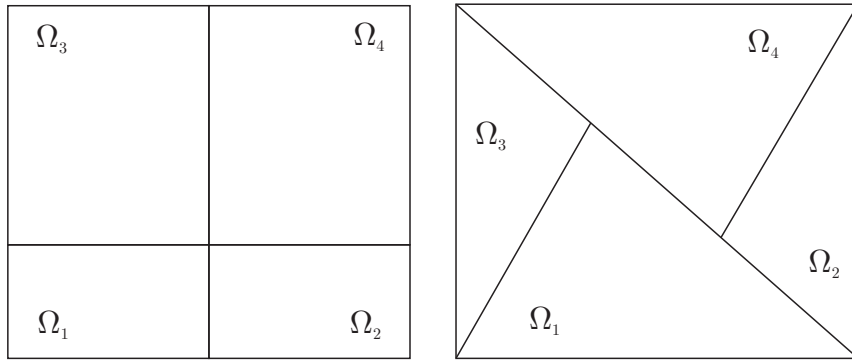
Přidáním (16) do soustavy (15) nám přibudou řádky v maticích $\mathbf{B}_1, \mathbf{B}_2$. Jednotlivé přidané řádky budou nulové vektory, kde na pozici odpovídající uzlu z \mathbf{u}_1^Γ vložíme 1 a na pozici odpovídající uzlu z \mathbf{u}_2^Γ vložíme -1. Toto provedeme pro všechny uzly na rozhraní. Publikace zabývající se použitím různých typů vazeb mezi podoblastmi je např. [13]. Vyřešením této rozšířené soustavy získáme řešení, které je spojitě na rozhraní.

2.3 Společná část

Metoda Mortarů společně s Internodes patří mezi metody rozložení oblasti (domain decomposition methods), při kterých se oblast na které je uvažována okrajová úloha rozdělí na $N \in \mathbb{N}$ podoblastí. V této práci se omezíme pouze na polygonální oblasti v rovině $\Omega \subset \mathbb{R}^2$. Pro každé dvě různé podoblasti platí, že se navzájem nepřekrývají a sjednocení jejich uzávěrů je uzávěr původní oblasti.

$$\bar{\Omega} = \bigcup_{i=1}^N \bar{\Omega}_i, \quad \Omega_i \cap \Omega_j = \emptyset \quad \text{pro } 1 \leq i \neq j \leq N.$$

Toto dělení oblasti může být geometricky konformní, nebo nekonformní. Geometricky konformní dělení je takové, kde pro každé dvě podoblasti platí, že průnik jejich uzávěrů je hrana, bod, nebo prázdná množina. V opačném případě jde o geometricky nekonformní dělení. Jak jde vidět na Obrázku 1, geometricky nekonformní dělení na rozdíl od konformního dovoluje mnohem větší volnost volby podoblastí.



Obrázek 1: Příklad geometricky konformního (vlevo) a nekonformního dělení (vpravo)

Tímto rozdělením vznikne kostra dekompozice, která může být později použita na rozdělení hran a hledání shodných vrcholů pro vytvoření rovnostních vazeb. Rozhraní Γ definujeme jako množinu bodů, které náležejí hranicím alespoň dvou podoblastí. Definici můžeme ekvivalentně napsat:

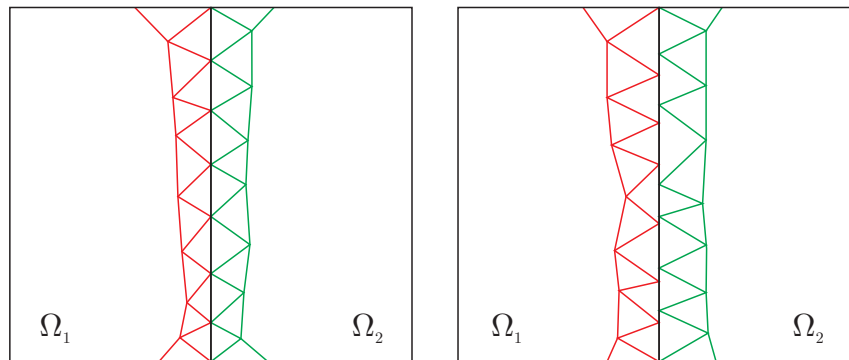
$$\Gamma = \overline{\bigcup_{i=1}^n \partial\Omega_i} \setminus \partial\Omega.$$

Rozhraním mezi dvěma oblastmi Ω_i, Ω_j je poté:

$$\Gamma_{i,j} = \bar{\Omega}_i \cap \bar{\Omega}_j.$$

Po rozdělení původní oblasti na podoblasti budeme pokračovat diskretizací jednotlivých podoblastí. Diskretizace spočívá v pokrytí oblasti konečnými prvky, v našem případě trojúhelníky. Jednou z vlastností diskretizovaných podoblastí je souhlasnost, nebo nesouhlasnost na rozhraní dvou sítí. Řekneme, že dvě sítě jsou souhlasné, pokud se na jejich rozhraní uzly obou sítí navzá-

jem překrývají. V opačném případě jde o nesouhlasné sítě, viz. Obrázek 2.



Obrázek 2: Příklad souhlasných (vlevo) a nesouhlasných sítí (vpravo)

Uzly na souhlasném rozhraní dvou souhlasných sítí můžeme „slepit“ přímo rovnostní vazbou $u_h^{(1)} = u_h^{(2)}$, kde $u_h^{(i)}$ představuje vektor odpovídající posunutí na rozhraní oblasti Ω_i . Rovnostní vazbu můžeme snadno zakomponovat do řešení pomocí Lagrangeových multiplikátorů a pohlížet na problém jako na sedlobodovou úlohu (11), viz. podkapitola 2.2.

Tato práce je zaměřena na nesouhlasné sítě, kde rovnost na rozhraní nelze vyřešit takto jednoduše. Jedním z přístupů výpočtu s nesouhlasnými sítěmi je právě použití Mortarů [5, 24, 25, 29], nebo Internodes [8, 14, 15].

Budeme pokračovat diskretizací výpočetní oblasti. Celý proces triangularizace a dělení na podoblasti může být realizováno několika způsoby:

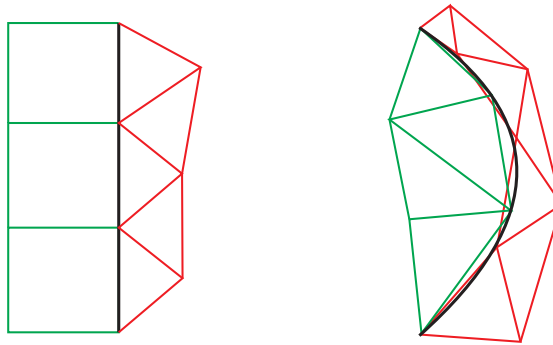
1. Triangularizace celé původní oblasti a následné dělení na podoblasti.
2. Dělení na podoblasti a souhlasná triangularizace na rozhraních (FETI).
3. Dělení na podoblasti a nezávislá triangularizace jednotlivých podoblastí.

Výsledek prvního a druhého způsobu je diskretizace, která je souhlasná na rozhraních mezi podoblastmi. Tento způsob zpracování ale probíhá z velké části sériově. Kvůli souhlasné triangularizaci je nutná komunikace při konstrukci sítí na rozhraní, to snižuje míru paralelizace procesu. V třetím způsobu lze naopak po rozdělení na podoblasti triangulovat zcela nezávisle, což je vhodné pro paralelní výpočty. Třetí z uvedených způsobů je poté asymptoticky nejrychlejší. Dovoluje snadno přizpůsobit síť předpokládanému řešení, nebo lokálně řešení zpřesnit. To může být velmi výhodné při výpočtu časově závislých úloh, kde například nemusí být možné předpovědět vývoj chování modelu. Nevýhoda tohoto přístupu jsou nesouhlasné sítě a s tím související složitější sestavení vazební matice.

Nesouhlasnost sítí společně s geometrickou nekonformitou jsou jen některé specifické druhy nekonformity. Přívlástek nekonformní obecně zastřešuje všechny možné druhy nekompatibility na rozhraní v metodách rozložení oblasti. Některé z typů nekonformity jsou

1. nesouhlasnost konečných prvků na rozhraní,
2. použití různých konečných prvků (jiné stupně, nebo typy),
3. geometrická nekonformnost,
4. nepřiléhající geometrie.

Nekonformita sítí je častá např. u kontaktních problémů, kde se předepisují podmínky nepronikání těles. V této práci se však omezíme pouze na rovnostní vazby na rozhraní, vhodné pro metody rozložení oblasti. Nicméně i v případě kontaktů lze odpovídající matice nepronikání sestavit obdobným způsobem jako v případě lepení. Na Obrázku 3 vlevo můžeme vidět použití různých konečných prvků. Poslední ze zmíněných typů nekonformity je zobrazen na Obrázku 3 vpravo, kde diskretizace podoblastí (rozlišené barvami) na sebe geometricky nepřiléhají.



Obrázek 3: Příklad použití různých konečných prvků (vlevo), nepřiléhající rozhraní (vpravo)

Primárně se budeme zabývat nesouhlasnými sítěmi a poté geometrickou nekonformitou. Z důvodu nesouhlasných uzlů na rozhraní nemůžeme vyžadovat bodovou spojitost. Místo toho musíme zvolit jinou, pokud možno ekvivalentní podmínku na rozhraní podoblastí tak, abychom došli ke srovnatelným výsledkům jako u souhlasných sítí.

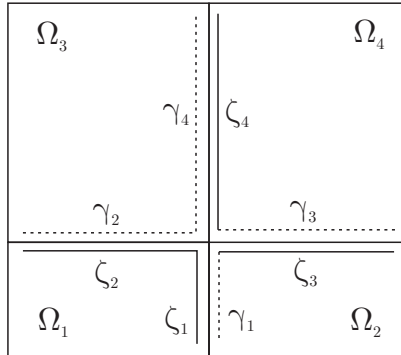
3 Mortar metoda konečných prvků

V této kapitole uvedeme a popíšeme metodu Mortarů, včetně implementace. Metoda Mortarů slouží ke spojení geometricky nekonformních i nesouhlasných sítí např. v metodě konečných prvků. Metoda může navíc sloužit ke spojení různých fyzikálních modelů, nebo různých druhů konečných prvků. Původně byla tato metoda poprvé popsána v práci [5] pro spektrální konečné prvky. Mortar metoda byla poté použita v různých aplikacích, například na *hp* konečné prvky [22], 3D spektrální prvky [1], nebo pro překrývající se dekompozice [6]. Jeden z novějších přístupů tzv. „new mortars“ jsou popsány v [28]. Metoda je stále aktuálním tématem a stále se jí věnuje řada prací.

3.1 Popis metody

Než přistoupíme k samotnému jádru metody, musíme provést ještě několik přípravných kroků. Prvním krokem je rozdělení hran podoblastí. Abychom byli v této práci konzistentní v označování hran, zavedeme následující označení. Hrany, případně strany budeme rozlišovat na master (řídící) a slave (podřízené). V literatuře věnující se tématu Mortarů (např. [5, 24]) jsou hrany typu master označovány jako mortar a hrany typu slave nonmortar.

Master hrany budeme označovat písmenem ζ a slave hrany písmenem γ . Na toto dělení hran budeme klást následující podmínku. Pro rozdělení hran platí, že pokud je hrana master resp. slave, tak poté protilehlá hrana je slave resp. master. Pravidlo lze intuitivně aplikovat na konformní dělení podoblastí, viz. Obrázek 4, kde master hrany jsou znázorněny plnou čarou a slave hrany přerušovanou čarou. Pro geometricky nekonformní dělení, postup popíšeme. Nejdříve



Obrázek 4: Příklad geometricky konformního dělení hran

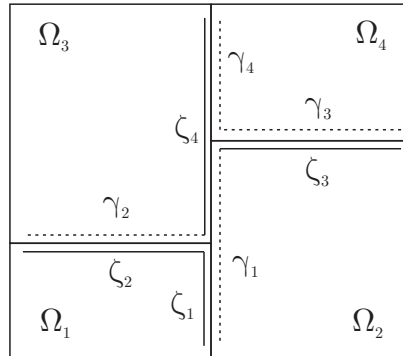
poznamenejme, že toto dělení hran obecně není jednoznačné, ale vždy existuje [24]. Protože dělení není jednoznačné, můžeme dojít k několika platným rozdělením hran. Tento fakt nijak neovlivňuje teorii Mortarů. Pokud bychom však měli konkrétní příklad, můžeme najít rozdíly ve výsledcích, viz. kapitola 5 Numerické experimenty. Možný postup je následující.

Algoritmus začne s libovolnou hranou, kterou označí jako master, poté hledá všechny protilehlé hrany. Protilehlé hrany označí jako slave hrany a zároveň je vloží do množiny zpracovávaných

hran \mathcal{F} . Dále se vybere hrana h z \mathcal{F} a hledají se k ní protilehlé neoznačené hrany, které se označí opačně než hrana h a vloží se do množiny \mathcal{F} . Poté se v cyklu prochází množina zpracovávaných hran dokud není prázdná. Postup je shrnut v Algoritmu 1.

Algorithm 1 Rozdělení hran

- 1: Začít s libovolnou hranou h na rozhraní Γ a označit ji jako master.
 - 2: Najít všechny protilehlé hrany k hraně h a ty označit jako slave a zároveň je vložit do množiny \mathcal{F} .
 - 3: **while** $\mathcal{F} \neq \emptyset$ **do**
 - 4: Vybrat libovolnou hranu $h \in \mathcal{F}$
 - 5: Najít všechny neoznačené protilehlé hrany k hraně h
 - 6: Pokud byla původní hrana h master, označit protilehlé jako slave
 - 7: Pokud byla původní hrana h slave, označit protilehlé jako master
 - 8: Vložit nově označené hrany do množiny \mathcal{F}
 - 9: Odebrat hranu h z množiny \mathcal{F}
 - 10: **end while**
-



Obrázek 5: Nekonformní příklad rozdělení hran

Tento postup zajistí, aby proti sobě neleželi konfigurace hran master-master, nebo slave-slave. Po rozdělení vzniknou dvě množiny hran $\hat{\zeta}$ a $\hat{\gamma}$. Obě množiny tvoří disjunktní rozdělení rozhraní Γ . Přirozeně platí, že sjednocení jejich uzávěrů pokryje celé rozhraní Γ . Symbolicky zapsáno

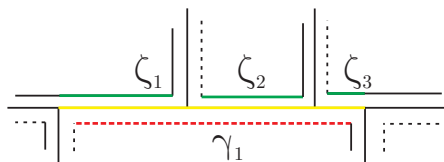
$$\Gamma = \bigcup_{i=1}^M \zeta_i, \quad \zeta_i \cap \zeta_j = \emptyset \quad \text{pro } 1 \leq i \neq j \leq |\hat{\zeta}|,$$

$$\Gamma = \bigcup_{i=1}^L \gamma_i, \quad \gamma_i \cap \gamma_j = \emptyset \quad \text{pro } 1 \leq i \neq j \leq |\hat{\gamma}|,$$

kde $|\hat{\zeta}|$ je počet prvků množiny $\hat{\zeta}$ a $|\hat{\gamma}|$ je počet prvků množiny $\hat{\gamma}$. Nyní přistoupíme k jádru mortar metody. Popíšeme situaci na jedné hraně náležící podoblasti, která patří do rozhraní Γ . Začneme pozorováním, že každá master hrana i slave hrana náleží právě jedné podoblasti. Protože obě množiny ζ i γ rozdělují celé rozhraní Γ , můžeme bez újmy na obecnosti zvolit slave hrany. Zvolme tedy libovolnou slave hranu γ_k . Označme oblast, které náleží hrana γ_k takto $\Omega_i^{(k)}$. Definujme Γ_k jako sjednocení všech částí mortar hran $\bar{\zeta}_{k,i}$, které se geometricky shodují s $\bar{\gamma}_k$, tj.

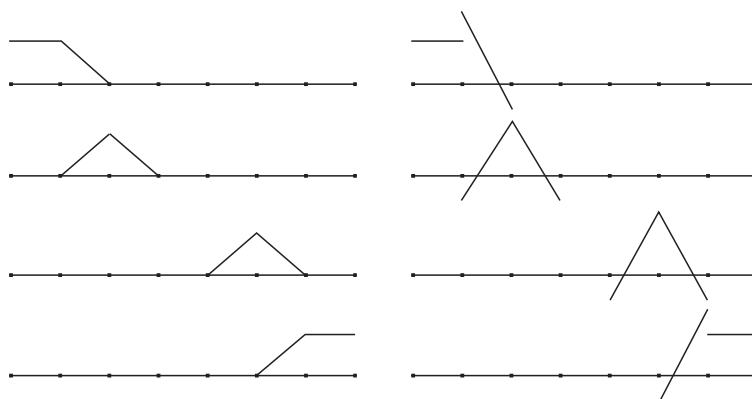
$$\Gamma_k = \bigcup_{i=1}^{s(\gamma_k)} (\bar{\zeta}_{k,i} \cap \bar{\gamma}_k).$$

Zavedené pojmy jsou znázorněny na Obrázku 6, kde je žlutou barvou vyznačena hrana Γ_k , červenou slave hrana a zelenou jsou vyznačeny všechny části master hran, které geometricky souvisí se slave hranou.



Obrázek 6: Příklad hrany Γ_k

Pro každou slave hranu zavedeme prostor testovacích funkcí $\Psi^h(\gamma_k)$, což je podprostor konečných prvků $V^h(\gamma_k)$, který je restrikcí $V^h(\Omega_i^{(k)})$ na γ_k . Pokud je prostor $V^h(\Omega_i^{(k)})$ tvořen po částech lineárními funkcemi, pak $\Psi^h(\gamma_k)$ je dán restrikcí $V^h(\Omega_i^{(k)})$ na γ_k s podmínkou, že tyto spojité po částech lineární funkce jsou konstantní v prvním a posledním intervalu sítě diskretizace na $\bar{\gamma}_k$ [24]. Použití jiných bázevých funkcí ke konstrukci prostoru $\Psi^h(\gamma_k)$ je popsáno v publikacích [28, 25]. Na Obrázku 7 jsou zobrazeny klasické i nové bázevové funkce popsané ve zmíněných pracích.



Obrázek 7: Vizualizace klasických (vlevo) a nových (vpravo) bázevých funkcí Ψ^h

Pro splnění podmínky (17) na γ_k potřebujeme popsat mortar projekci π_{q_1, q_2} . Tato projekce zobrazí funkci $u_k \in L^2(\Gamma_k)$ na $V^h(\gamma_k)$ tak, že pro dvě hodnoty q_1, q_2 se funkce $\pi_{q_1, q_2}(u_k)$ se rovná těmto hodnotám na koncích γ_k .

Přes každou slave stranu budeme vyžadovat splnění podmínky slabé spojitosti pro funkci u . Nechť u_{γ_k} je restrikce u na γ_k a u_{Γ_k} je analogicky restrikce u na Γ_k . Funkce u na γ_k je poté dána tzv. mortar podmínkou

$$\int_{\gamma_k} (u_k - \pi_{q_1, q_2}(u_k))\psi \, ds = 0, \quad \forall \psi \in \Psi^h(\gamma_k). \quad (17)$$

Vnitřní uzly slave stran nesouvisí se stupni volnosti konečně prvkového prostoru V^h , ale hodnoty funkce u na jejích koncích ano. Každá funkce $u \in V^h$, je nulová ve všech bodech $\partial\Omega_D$ a zároveň restrikce na podoblast Ω_i je klasická konečně prvková funkce. Po funkci u budeme požadovat, aby splnila mortar podmínku (17) pro každou slave hranu γ_k . Jinými slovy mortar podmínka požaduje, aby restrikce u na $\bar{\gamma}_k$ byla rovna projekci u na Γ_k , tj.

$$u|_{\bar{\gamma}_k} = \pi_{u_A, u_B}(u|_{\Gamma_k}),$$

kde u_A, u_B jsou hodnoty u v koncových bodech A, B hrany $\bar{\gamma}_k$. Pro více informací např. o stabilitě mortar projekce odkazujeme na text [24].

3.2 Algebraická formulace mortar podmínky

Než představíme algebraickou podobu mortar podmínky (17), zavedeme následující označení. Nechť γ_k je libovolná slave hrana, \mathbf{u}_{γ_k} je vektor hodnot vnitřních uzlů na hraně γ_k . Dále nechť \mathbf{u}_{Γ_k} je vektor hodnot na koncích γ_k a také hodnot odpovídajícím mortar uzlům z master hrany patřícím do Γ_k . Potom můžeme \mathbf{u}_{γ_k} jednoznačně určit podle \mathbf{u}_{Γ_k} a mortar podmínky (17). Napsáno v maticovém zápisu

$$\mathbf{M}\mathbf{u}_{\gamma_k} - \mathbf{D}\mathbf{u}_{\Gamma_k} = \mathbf{o}, \quad (18)$$

můžeme vyjádřit \mathbf{u}_{γ_k}

$$\mathbf{u}_{\gamma_k} = \mathbf{M}^{-1}\mathbf{D}\mathbf{u}_{\Gamma_k} = \mathbf{P}\mathbf{u}_{\Gamma_k}. \quad (19)$$

Matice \mathbf{M} a \mathbf{D} nazýváme mortar matice. Kombinace mortar matic $\mathbf{P} = \mathbf{M}^{-1}\mathbf{D}$ představuje diskrétní podobu mortar projekce. Mortar podmínku zahrneme do sedlobodové formulace (14) tak, že přidáme matice \mathbf{M} a \mathbf{D} do matice vazeb \mathbf{B} . Pro ilustraci popíšeme opět situaci pro dvě podoblasti. V tomto případě rozšíříme matice $\mathbf{B}_1, \mathbf{B}_2$ a vektory $\boldsymbol{\lambda}, \mathbf{c}$ takto

$$\hat{\mathbf{B}}_1 = \begin{bmatrix} \mathbf{B}_1 \\ \mathbf{M} \end{bmatrix}, \quad \hat{\mathbf{B}}_2 = \begin{bmatrix} \mathbf{B}_2 \\ -\mathbf{D} \end{bmatrix}, \quad \hat{\boldsymbol{\lambda}} = \begin{bmatrix} \boldsymbol{\lambda} \\ \boldsymbol{\mu} \end{bmatrix}, \quad \hat{\mathbf{c}} = \begin{bmatrix} \mathbf{c} \\ \mathbf{o} \end{bmatrix}.$$

Výsledná soustava pro dvě podoblasti je

$$\begin{bmatrix} \mathbf{A}_1 & \mathbf{O} & \mathbf{B}_1^T & \mathbf{M}^T \\ \mathbf{O} & \mathbf{A}_2 & \mathbf{B}_2^T & -\mathbf{D}^T \\ \mathbf{B}_1 & \mathbf{B}_2 & \mathbf{O} & \mathbf{O} \\ \mathbf{M} & -\mathbf{D} & \mathbf{O} & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \boldsymbol{\lambda} \\ \boldsymbol{\mu} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{c} \\ \mathbf{o} \end{bmatrix} \iff \begin{bmatrix} \mathbf{A}_1 & \mathbf{O} & \hat{\mathbf{B}}_1^T \\ \mathbf{O} & \mathbf{A}_2 & \hat{\mathbf{B}}_2^T \\ \hat{\mathbf{B}}_1 & \hat{\mathbf{B}}_2 & \mathbf{O} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \hat{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \hat{\mathbf{c}} \end{bmatrix}. \quad (20)$$

Pokud budeme mít v dekompozici tři a více podoblastí může se vyskytnout kromě nesouhlasných sítí i geometrická nekonformita.

3.3 Implementace

V této kapitole se dostáváme k praktické části práce. Popíšeme zde algoritmy, které byly implementovány v programovém prostředí MATLAB a jsou součástí přílohy této práce. Následující text vychází z práce [29]. Splnění mortar podmínky (17) ekvivalentně vynutíme pomocí mortar matic \mathbf{D} , \mathbf{M}

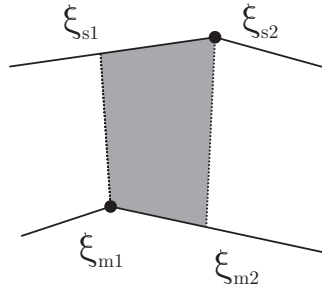
$$\mathbf{D}(i, j) = \int_{\gamma_k} N_i(\xi^{(1)}(x)) N_j(\xi^{(1)}(x)) ds \quad (21a)$$

$$\mathbf{M}(i, k) = \int_{\gamma_k} N_i(\xi^{(1)}(x)) N_k(\xi^{(2)}(\chi(x))) ds. \quad (21b)$$

Oba integrály (21a), (21b) jsou definovány na slave hraně $\gamma_k = \Gamma_k$. Z důvodu, že se v integrálu (21b) se vyskytují výrazy z obou stran hrany, musíme oblast integrace rozdělit. Důvodem je samozřejmě nekonformita na rozhraní sítí. Způsob dělení γ_k musí být takový, aby příspěvky z obou stran byly hladké ve smyslu normálových směrů. Dělení musí tedy minimálně zahrnovat body diskretizace z obou stran rozhraní. Toto dělení vede ke konstrukci tzv. segmentů.

3.3.1 Projekce a segmentace

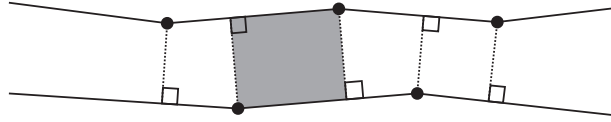
Pod pojmem segment rozumíme čtyřúhelníkovou oblast s vrcholy v bodech ξ_{s1} , ξ_{s2} , ξ_{m1} , ξ_{m2} . Příklad segmentu je na Obrázku 8 vyznačen šedou barvou.



Obrázek 8: Segment

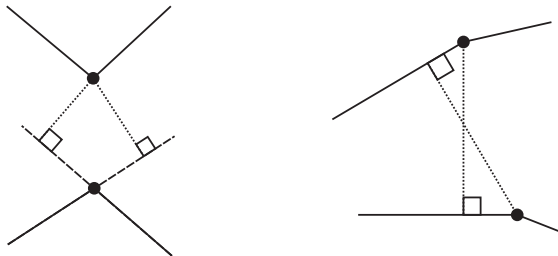
Základem konstrukce segmentů je projekce z master hrany na slave hranu a naopak. Při tradičním

způsobu segmentace se uzly na rozhraní projektují na opačnou hranu tak, že se hledá nejkratší kolmice mezi uzlem a protější hranou nebo bodem.



Obrázek 9: Tradiční segmentace

Značná nevýhoda tohoto způsobu je, že může dojít k nerozhodnutelným situacím. Na Obrázku 10 je možno vidět, že pokud budeme hledat projekci mezi uzly na rohu diskretizace, tak ji tradičním přístupem nenajdeme. Příčina tohoto problému je nespojitost normál v uzlech diskretizace. Dalším problémem může být křížení normál.



Obrázek 10: Nedokonalost tradiční segmentace

Poznámka 3 V následujícím textu pod pojmem element myslíme hranu mezi uzly diskretizace A, B , označujeme e_{AB} . Důvodem je rozlišení od pojmu hrana.

Ukážeme zde více robustní segmentační metodu, která je popsána v práci [29]. Předpokládejme, že máme diskrétní podobu mortar rozhraní. Vyberme hranu γ_k odpovídající slave hraně. Geometrie této diskrétní hrany má obecně skokové změny ve směru vnější normály. Jak jsme výše ukázali, tato vlastnost může způsobit během vyhledávání segmentů potíže. Aby k tomu nedocházelo musíme zavést pojmy jako normálové směry, které zvýší robustnost algoritmu vyhledávání segmentů. Normálové směry budou definovány v uzlech hrany γ_k . Protože k samostatnému bodu nelze přiřadit normálu, budeme uzlovým bodům přiřazovat normálu podle elementů, které jsou s tímto bodem sousedí. Každý uzel u_A náleží dvěma sousedním elementům e_{BA} a e_{AC} , s výjimkou koncových uzlů, které sousedí jen s jedním elementem. Z toho plyne, že pro vnitřní uzly máme obecně dva různé normálové směry. Aby normálové směry byly dány jednoznačně, bude normála v bodě diskretizace definována jako vážený průměr normál přilehlých elementů tohoto uzlu. Vytvoření normálového směru v bodě A je vizualizováno na Obrázku 11.

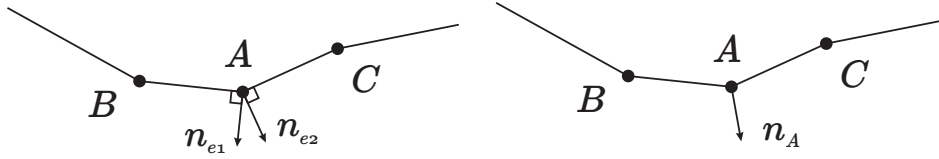
Definice 4 (normálový směr) *Necht A je uzel diskretizace na mortar rozhraní. Potom definujeme normálový směr pro každý uzel takto:*

$$\mathbf{n}_A = \frac{\hat{\mathbf{n}}_A}{\|\hat{\mathbf{n}}_A\|},$$

kde $\hat{\mathbf{n}}_A$ je

$$\hat{\mathbf{n}}_A = \begin{cases} |e_1| \mathbf{n}_{e_1} + |e_2| \mathbf{n}_{e_2} & A \in \Gamma \setminus \partial\Gamma \\ \mathbf{n}_{e_1} & A \in \partial\Gamma \end{cases},$$

kde $|e_i|$ je délka hrany e_i .



Obrázek 11: Normálový směr

Dále budeme definovat normálové pole, které bude „spojitým rozšířením“ normálových směrů na celou hranu γ_k .

Definice 5 (normálové pole)

$$\mathbf{n}_{\gamma_k}(\xi) = \sum_A N_A(\xi) \mathbf{n}_A,$$

kde \mathbf{n}_A je normálový směr a N_A je příslušná tvarová funkce uzlu A .

Nyní popíšeme obě projekce nutné k segmentaci. Začneme projekcí ze slave na master hranu. Projekce spočívá ve výpočtu průsečíku přímky vedené slave uzlem ve směru vnější normály a master elementem. K výpočtu použijeme rovnici

$$\left[N_1(\xi^{(2)}) \mathbf{x}_{m_1} + N_2(\xi^{(2)}) \mathbf{x}_{m_2} - \mathbf{x}_s \right] \cdot \mathbf{n}_s = 0, \quad (22)$$

kterou budeme řešit pro neznámou $\xi^{(2)}$, která má význam souřadnice projekce slave uzlu na master element. Proměnné \mathbf{x}_{m_1} , \mathbf{x}_{m_2} jsou vektory souřadnic uzlů na master elementu, \mathbf{x}_s je vektor souřadnic uzlu na slave straně rozhraní. Funkce N_1 , N_2 jsou příslušné tvarové funkce (shape functions) master uzlů. Vektor \mathbf{n}_s je normála příslušející slave uzlu.

O něco složitější je projekce z master na slave stranu. I nadále uvažujeme pouze normálové pole ze slave strany. Projekci hledáme analogicky mezi master uzlem a hranou mezi dvěma slave

uzly. Směr této projekce bude z normálového pole na slave straně. Řešením rovnice

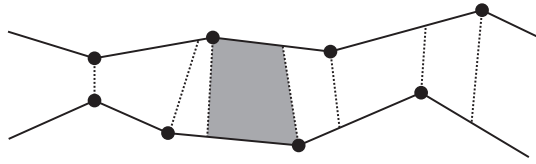
$$\left[N_1(\xi^{(1)})\mathbf{x}_{s_1} + N_2(\xi^{(1)})\mathbf{x}_{s_2} - \mathbf{x}_m \right] \cdot \left[N_1(\xi^{(1)})\mathbf{n}_{s_1} + N_2(\xi^{(1)})\mathbf{n}_{s_2} \right] = 0 \quad (23)$$

vypočteme $\xi^{(1)}$, což je souřadnice projekce master uzlu na slave elementu. Proměnná \mathbf{x}_m představuje vektor souřadnic master uzlu, \mathbf{x}_{s_1} , \mathbf{x}_{s_2} jsou souřadnice uzlů na slave elementu. Vektory \mathbf{n}_{s_1} , \mathbf{n}_{s_2} jsou normály v bodech \mathbf{x}_{s_1} , \mathbf{x}_{s_2} . Funkce N_1 , N_2 jsou tvarové funkce příslušející slave uzlům.

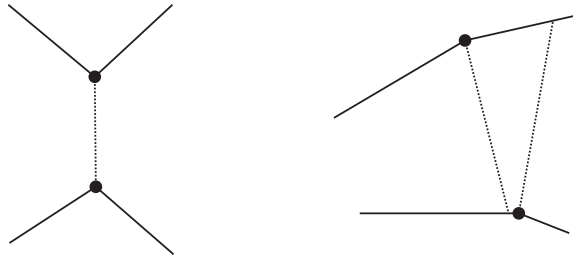
Poznámka 4 Pro lineární konečné prvky je (22) lineární rovnice a (23) je kvadratická rovnice. V případě že se normály \mathbf{n}_{s_1} a \mathbf{n}_{s_2} rovnají, je i (23) lineární rovnicí. Konkrétně pro lineární prvky jsou tvarové funkce následující

$$N_1(\xi) = \frac{1 - \xi}{2}, \quad N_2(\xi) = \frac{1 + \xi}{2}. \quad (24)$$

Pokud proměnná ξ náleží do intervalu $\langle -1, 1 \rangle$, tak se projekce uzlu zobrazila na protilehlý element, nebo na jeho krajní uzly pro $|\xi| = 1$. V opačném případě je absolutní hodnota $|\xi| > 1$ a projekce zobrazila uzlu mimo element. Výsledek robustní segmentace je na Obrázku 12, kde je šedou barvou naznačena oblast jednoho segmentu. Zároveň můžeme vidět na Obrázku 13 jak se tento přístup vyrovnal s nedokonalostmi tradiční segmentace.



Obrázek 12: Výsledná robustní segmentace



Obrázek 13: Robustní segmentace

Algoritmus, který se chystáme popsat najde a zpracuje všechny segmenty, které potom uloží do datové struktury. V této datové struktuře bude uložen pod indexem slave elementu seznam všech příspěvků z master elementů, které se slave elementem souvisí. Protože mortar integrály

(21a), (21b) jsou přes hranu typu slave, je vhodné uvažovat makro segmenty, které budou obsahovat segmenty, které mají společný slave element. Makro segmenty chápeme jako sjednocení více segmentů. Výsledkem algoritmu bude datová struktura, která bude obsahovat informace o všech makro segmentech. Nyní přistoupíme k samotnému algoritmu segmentace. Algoritmus potřebuje na vstupu seznam všech elementů E , indexy master elementů I_m a slave elementů I_s , normály v bodech na rozhraní N . Procedury *slave2master* a *master2slave* představují algoritmy řešící rovnice (22) a (23). Tyto procedury navíc potřebují na vstupu souřadnice zadaných uzlů. Implementace těchto procedur je ve stejnojmenných souborech *slave2master.m* a *master2slave.m*.

Algorithm 2 Segmentace

Require: I_s, I_m, E, N

```

1: for  $i$  in  $I_s$  do
2:    $s = E(i)$ 
3:   for  $j$  in  $I_m$  do
4:      $m = E(j)$ 
5:      $p_1 = \text{slave2master}(s_1, m_1, m_2, N(s_1))$ 
6:      $p_2 = \text{slave2master}(s_2, m_1, m_2, N(s_1))$ 
7:     if  $p_1 > 1$  AND  $p_2 > 1$  then
8:       next j
9:     end if
10:    if  $p_1 < -1$  AND  $p_2 < -1$  then
11:      next j
12:    end if
13:     $p_1 = \text{master2slave}(m_1, s_1, s_2, N(s_1), N(s_2))$ 
14:     $p_2 = \text{master2slave}(m_2, s_1, s_2, N(s_1), N(s_2))$ 
15:     $p_1 = \text{limitize}(p_1)$ 
16:     $p_2 = \text{limitize}(p_2)$ 
17:    if  $\|p_1 - p_2\| < \varepsilon$  then
18:      next j
19:    end if
20:    push( $S(i), [j; p_1; p_2]$ )
21:  end for
22: end for

```

Algoritmus má dva vnořené cykly; vnější přes slave elementy a vnitřní přes master elementy. Ve vnitřním cyklu se spočte projekce z uzlů aktuálního slave elementu s na aktuální mortar element m . Podle výsledku p_1, p_2 se určí, jestli se projekce ze slave elementu s zobrazila na nějaké části mortar elementu m . Pokud ano, tak se spočte zpětná projekce z uzlů aktuálního master elementu m na slave element s . Pokud hodnota nějakého výsledků zpětné projekce bude

v absolutní hodnotě větší než 1, tak se nastaví na $p_i = \text{sgn}(p_i)$ (funkce *limitize.m*). Pro případ, že by rozdíl hodnot zpětných projekcí byl téměř nulový, tak se příspěvek neuloží (oblast nulové míry). V opačném případě se do datové struktury S pod indexem slave elementu i uloží spolu s indexem master elementu j hodnoty zpětné projekce p_1, p_2 . Pokud se najde další příspěvek související s tím samým slave elementem s , je přidán k předchozím. Takto se najdou všechny příspěvky z mortar elementů na vybraný slave element. Struktura S ve výsledku obsahuje ke každému slave elementu seznam všech master příspěvků, které se slave elementem souvisí. Tato struktura obsahuje informace potřebné k určení makro segmentů.

3.3.2 Sčítání příspěvků ze segmentů

Jak jsme již zmínili výše, integrály (21a), (21b) vypočteme jako součet integrálů přes jednotlivé segmenty, nebo ekvivalentně přes makro segmenty

$$D_{i,j} = \sum_{\text{seg}} D_{i,j}^{\text{seg}} = \sum_{\text{seg}} \int_{\gamma_k^{\text{seg}}} N_i(\xi^{(1)}(x)) N_j(\xi^{(1)}(x)) ds, \quad (25a)$$

$$M_{i,j} = \sum_{\text{seg}} M_{i,j}^{\text{seg}} = \sum_{\text{seg}} \int_{\gamma_k^{\text{seg}}} N_i(\xi^{(1)}(x)) N_k(\xi^{(2)}(\chi(x))) ds. \quad (25b)$$

Díky segmentaci již nemusíme použít mapování χ . K výpočtu je potřeba ještě zavést parametrizaci segmentu. Proměnná η bude v rozmezích -1 až 1 v rámci každého segmentu. Abychom dali do souvislosti souřadnice mezi body segmentu a parametrizací η , zavedeme interpolaci

$$\xi^{(i)} = \frac{1-\eta}{2} \xi_1^{(i)} + \frac{1+\eta}{2} \xi_2^{(i)}.$$

Nyní můžeme formálně napsat tvar příspěvků k mortar integrálům (25a),(25b) za segment

$$D_{i,j}^{\text{seg}} = \int_{-1}^1 N_i(\xi^{(1)}(\eta)) N_j(\xi^{(1)}(\eta)) \left\| \frac{\partial \varphi_e}{\partial \xi^{(1)}} \right\| \frac{\partial \xi^{(1)}}{\partial \eta} d\eta, \quad (26a)$$

$$M_{i,k}^{\text{seg}} = \int_{-1}^1 N_i(\xi^{(1)}(\eta)) N_k(\xi^{(2)}(\eta)) \left\| \frac{\partial \varphi_e}{\partial \xi^{(1)}} \right\| \frac{\partial \xi^{(1)}}{\partial \eta} d\eta. \quad (26b)$$

Nahradíme-li integraci numerickou kvadraturou, vyjdou nám následující aproximace

$$D_{i,j}^{\text{seg}} \approx \sum_{g=1}^q w_r N_i(\xi^{(1)}(\eta_g)) N_j(\xi^{(1)}(\eta_g)) J, \quad (27a)$$

$$M_{i,k}^{\text{seg}} \approx \sum_{g=1}^q w_r N_i(\xi^{(1)}(\eta_g)) N_k(\xi^{(2)}(\eta_g)) J, \quad (27b)$$

kde η_r jsou souřadnice kvadraturních bodů, w_r jsou příslušné váhy a J je jakobián segmentu

$$J = \left\| \frac{\partial \varphi_e^{(1)}}{\partial \eta} \right\| = \left\| \frac{\partial \varphi_e^{(1)}}{\partial \xi^{(1)}} \right\| \frac{\partial \xi^{(1)}}{\partial \eta}.$$

V naší konkrétní implementaci pro lineární prvky je jakobián roven

$$J = \frac{\|\mathbf{x}_{s_1} - \mathbf{x}_{s_2}\|}{2}.$$

V implementaci numerického výpočtu mortar integrálů (25a), (25b) používáme třibodovou kvadraturu. Výpočet integrálů na segmentech je implementován v souboru *calc_mortar_matrices.m*. Nyní zbývá jen vložit vypočtené hodnoty mortar integrálů přes všechny segmenty do matic \mathbf{D} a \mathbf{M} . Programová funkce realizující všechny kroky potřebné k vygenerování mortar matic je uložena v souboru *mortar.m*.

4 Metoda Internodes

V této kapitole popíšeme metodu Internodes [8, 14, 15] a její implementaci. Internodes je zkratka pro „interpolation for nonconforming decompositions“, v překladu jde o interpolaci pro nekonformní dekompozice. Metoda Internodes byla publikována v práci [8] pro nekonformní numerické řešení eliptických okrajových úloh druhého stupně. Její aplikace byly rozšířeny například na Navier-Stokesovi rovnice. Internodes bývají označovány za alternativu k výše popsané Mortar metodě. Podobě jako u Mortarů dokáže tato metoda překlenout více typů nekonformity.

Opět se v této kapitole primárně zaměříme na nesouhlasné sítě, a poté na geometrickou nekonformitu. Hlavní myšlenkou zde bude, vytvořit způsob, který zprostředkuje komunikaci mezi nekonformními sítěmi. Klíčem k této komunikaci bude interpolace, která je přirozenou volbou pro překlenutí nekompatibility sítí.

4.1 Popis metody

Následující text vychází z publikovaných prací [8, 14, 15]. V této podkapitole stručně popíšeme diskrétní podobu okrajové úlohy (1). Rovnou budeme uvažovat dělení výpočetní oblasti na dvě nepřekrývající se podoblasti Ω_1 a Ω_2 .

Uvažujme množinu \mathcal{T}_{k,h_k} , která představuje diskretizační síť (triangulaci) podoblasti Ω_k s kladným parametrem sítě h_k . Diskretizaci celé oblasti Ω značíme $\mathcal{T}_h = \bigcup_k \mathcal{T}_{k,h}$. Předpokládáme, že diskretizace \mathcal{T}_h je afinní, regulární a kvazi-uniformní¹. Ještě bychom měli uvést prostor \mathbb{P}_p polynomů stupně nejvýše p . Pro naše účely se v této práci omezíme na lineární trojúhelníkové prvky, tedy \mathbb{P}_1 . Uvažujme prostor diskretizace oblasti

$$X_{k,h} = \{v \in C^0(\overline{\Omega_k}) : v|_T \in \mathbb{P}_1, \forall T \in \mathcal{T}_{k,h_k}\}, \quad (28)$$

který obsahuje spojité funkce na $\overline{\Omega_k}$, jejichž restrikce na prvcích diskretizace je lineární funkcí. Dále představíme konečně prvkové podprostory $V_{k,h}$ a $V_{k,h}^0$ prostoru $V_k = V(\Omega_k)$

$$V_{k,h} = X_{k,h} \cap V_k, \quad V_{k,h}^0 = \{v \in V_{k,h} : v|_\Gamma = 0\}. \quad (29)$$

Pokračujeme definicí prostorů stop

$$\Lambda_{k,h} = \{\lambda = v|_\Gamma : v \in X_{k,h}\}, \quad \Lambda_{k,h}^0 = \{\lambda \in \Lambda_{k,h} : \lambda|_{\partial\Gamma} = 0\} \subset \Lambda_{k,h}. \quad (30)$$

Označme dimenze těchto prostorů

$$N_k = \dim(\Lambda_{k,h}), \quad N_k^0 = \dim(\Lambda_{k,h}^0), \quad \hat{N}_k = \dim(V_{k,h}). \quad (31)$$

¹Tyto vlastnosti jsou popsány v [7]

Bázové funkce prostoru $V_{k,h}$ příslušející uzlům diskretizace $x_i^{(k)} \in \mathcal{T}_{k,h_k}$ označíme jako $\{\varphi_i^{(k)}\}$, $i = 1, \dots, \tilde{N}_k$. Bázové funkce prostoru $\Lambda_{k,h}$ označíme jako $\{\phi_i^{(k)}\}$, $i = 1, \dots, N_k$. Poznamenejme, že prostor $\Lambda_{k,h}^0$ na rozdíl od $\Lambda_{k,h}$ je definován tak, aby nezahrnoval uzly související s Dirichletovou okrajovou podmínkou. Důvodem je, že tyto uzly již nejsou stupni volnosti, protože jsou na nich hodnoty řešení a-priori dány. Pokud jsou některé uzly z rozhraní Γ současně i uzly hranice oblasti $\partial\Omega$ a patří všechny tyto uzly do $\partial\Omega_N$, tak prostory $\Lambda_{k,h}$ a $\Lambda_{k,h}^0$ jsou totožné. V opačném případě jsou v množině $\Gamma \cap \partial\Omega$ uzly, které náležejí také $\partial\Omega_D$ a nejsou zahrnuty v prostoru $\Lambda_{k,h}^0$, lépe řečeno $\Lambda_{k,h}^0$ neobsahuje bázové funkce související s uzly $\Gamma \cap \partial\Omega_D$. To také znamená, že $N_k^0 \leq N_k$. Pro usnadnění značení budeme předpokládat, že N_k^0 bázových funkcí $\Lambda_{k,h}^0$ se shodují s prvními N_k^0 bázovými funkcemi $\Lambda_{k,h}$.

4.2 Interpolace

Budeme pokračovat popisem dvou nezávislých interpolačních operátorů, které jsou základním kamenem této metody. Operátory

$$\Pi_{12} : \Lambda_{2,h} \rightarrow \Lambda_{1,h}, \quad \Pi_{21} : \Lambda_{1,h} \rightarrow \Lambda_{2,h} \quad (32)$$

budou sloužit k výměně informací mezi nezávisle diskretizovanými podoblastmi na Γ . Tyto operátory můžeme chápat jako klasické Lagrangeovy interpolační operátory, které nyní popíšeme. Necht $\lambda_{1,h}$ je funkcí z prostoru $\Lambda_{1,h}$, rozepíšeme ji vzhledem k bázi $\{\phi_i^{(2)}\}$ prostoru $\Lambda_{2,h}$

$$(\Pi_{21}\lambda_{1,h})(x) = \sum_{i=1}^{N_2} (\Pi_{21}\lambda_{1,h})(x_i^{(\Gamma_2)})\phi_i^{(2)}(x), \quad \forall x \in \Gamma_2. \quad (33)$$

Funkci dále rozepíšeme vzhledem k její vlastní bázi

$$\lambda_{1,h}(x) = \sum_{j=1}^{N_1} \lambda_{1,h}(x_j^{(\Gamma_1)})\phi_j^{(1)}, \quad \forall x \in \Gamma_1. \quad (34)$$

Pak pro každé $x \in \Gamma_2$

$$(\Pi_{21}\lambda_{1,h})(x) = \sum_{i=1}^{N_2} \left(\sum_{j=1}^{N_1} \lambda_{1,h}(x_j^{(\Gamma_1)}) (\Pi_{21}\phi_j^{(1)})(x_i^{(\Gamma_2)}) \right) \phi_i^{(2)}(x). \quad (35)$$

Analogickou úvahu použijeme i pro operátor Π_{12} . Interpolační matice \mathbf{R}_{21} resp. \mathbf{R}_{12} související s operátory Π_{21} resp. Π_{12} jsou definovány takto

$$(\mathbf{R}_{21})_{i,j} = (\Pi_{21}\phi_j^{(1)})(x_i^{(\Gamma_2)}), \quad i = 1, \dots, N_2, \quad j = 1, \dots, N_1, \quad (36)$$

$$(\mathbf{R}_{12})_{i,j} = (\Pi_{12}\phi_j^{(2)})(x_i^{(\Gamma_1)}), \quad i = 1, \dots, N_1, \quad j = 1, \dots, N_2. \quad (37)$$

V situaci, kdy je rozhraní mezi podoblastmi přiléhající (viz. podkapitola 2.3), použijeme již zmíněné Lagrangeovy interpolační operátory. Pro nepřiléhající geometrie použijeme radiální bázové funkce a s nimi související interpolační operátory. Konkrétně popíšeme Rescaled Localized Radial Basis Functions, zkráceně RL-RBF, kterým se věnuje publikace [4]. Uvedme zde nejdříve Wendlandovy [27] radiální bázové funkce s kompaktním nosičem se středem v uzlech $x_i^{(\Gamma_k)}$ s poloměrem $r > 0$

$$\tilde{\phi}_i^{(k)}(x) = \max \left\{ 0, \left(1 - \frac{\|x - x_i^{(\Gamma_k)}\|}{r} \right)^4 \right\} \left(1 + \frac{4\|x - x_i^{(\Gamma_k)}\|}{r} \right), \quad (38)$$

kde $\|\cdot\|$ je eukleidovská norma. Bázová funkce je vykreslena na Obrázku 14. Tyto bázové funkce jsou použity při konstrukci RL-RBF operátorů. Nyní rovnou přejdeme k operátorům interpolace generované RL-RBF $\Pi_{21} : \Lambda_{1,h} \rightarrow \Lambda_{2,h}$ a $\Pi_{12} : \Lambda_{2,h} \rightarrow \Lambda_{1,h}$. Operátory jsou definovány pro každou $\lambda_{1,h} \in \Lambda_{i,h}$ a $\lambda_{2,h} \in \Lambda_{2,h}$

$$(\Pi_{21}\lambda_{1,h})(x_i^{(\Gamma_2)}) = (\Pi_{RL-RBF}^{(1)}\lambda_{1,h})(x_i^{(\Gamma_2)}), \quad i = 1, \dots, N_1, \quad (39)$$

$$(\Pi_{12}\lambda_{2,h})(x_i^{(\Gamma_1)}) = (\Pi_{RL-RBF}^{(2)}\lambda_{2,h})(x_i^{(\Gamma_1)}), \quad i = 1, \dots, N_2. \quad (40)$$

Rovnice pro interpolační matice související s RL-RBF jsou

$$(\mathbf{R}_{21})_{i,j} = (\Pi_{21}\phi_j^{(1)})(x_i^{(\Gamma_2)}) = \frac{(\Phi_{21}\Phi_{11}^{-1})_{ij}}{(\Phi_{21}\Phi_{11}^{-1}\mathbf{1})_i}, \quad i = 1, \dots, N_2, \quad j = 1, \dots, N_1 \quad (41)$$

$$(\mathbf{R}_{12})_{i,j} = (\Pi_{12}\phi_j^{(2)})(x_i^{(\Gamma_1)}) = \frac{(\Phi_{12}\Phi_{22}^{-1})_{ij}}{(\Phi_{12}\Phi_{22}^{-1}\mathbf{1})_i}, \quad i = 1, \dots, N_1, \quad j = 1, \dots, N_2, \quad (42)$$

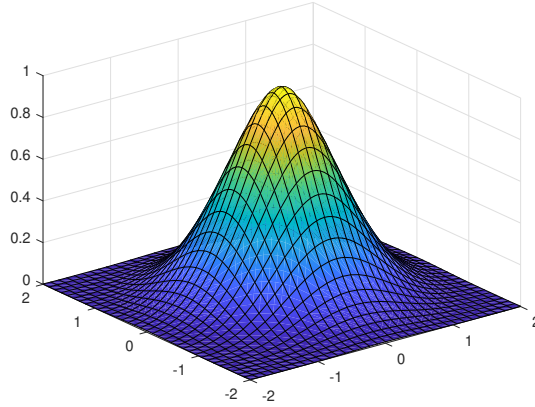
kde $\mathbf{1}$ je sloupcový vektor jedniček a matice

$$(\Phi_{kl})_{ij} = \tilde{\phi}_j^{(l)}(x_i^{(\Gamma_k)}), \quad k, l \in \{1, 2\}, \quad i = 1, \dots, N_k, \quad j = 1, \dots, N_l. \quad (43)$$

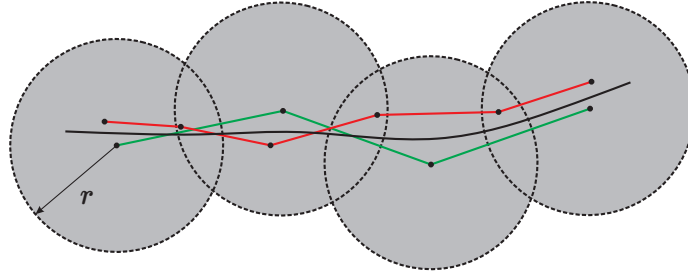
K výpočtu matic \mathbf{R}_{21} a \mathbf{R}_{12} jsou potřeba jen souřadnice bodů z obou hran rozhraní. Postup odvození je uveden například v [15]. Poznamenejme, že pro oba případy interpolace platí, že pro konformní případ jsou obě interpolační matice \mathbf{R}_{21} , \mathbf{R}_{12} čtvercové a jsou rovny jednotkové matici $\mathbf{R}_{21} = \mathbf{R}_{12} = \mathbf{I}$.

Obrázek 14 ilustruje tvar jedné radiální bázové funkce ve 2D. Na Obrázku 15 můžeme vidět aplikaci RBF na uzly nepřiléhajících diskretizací, kde červenou a zelenou barvou jsou rozlišeny hrany podoblastí na rozhraní Γ , které je naznačeno černou čarou a šedě je znázorněné sjednocení nosičů RBF. Zároveň můžeme pozorovat, že volba poloměru r bude záviset na geometrii rozhraní. Optimálně by poloměr měl být co nejmenší, protože čím více má RBF menší nosič tak, tím jsou interpolační matice více řídké. Na druhou stranu by poloměr měl být dostatečně velký, aby

zahrnoval alespoň dva uzly z protějšší hrany. Více o optimální volbě poloměru v [4].



Obrázek 14: Radiální bázová funkce v bodě $(0, 0)$ s poloměrem $r = 2$



Obrázek 15: Interpolace uzlů sítí pomocí RBF

4.3 Formulace

Pro obě podoblasti definujeme diskrétní spojitý lineární operátor

$$E_k : \Lambda_h \rightarrow X_{k,h} \text{ tak, že } (E_k \lambda_h)|_{\Gamma} = \lambda_h. \quad (44)$$

Operátor E_k rozšiřuje stopu λ_h na $V_{k,h}$ tak, že funkce $E_k \lambda_h$ je nulová ve všech uzlech diskretizace $\mathcal{T}_{k,h}$. Například pro každou bázovou funkci $\phi_j^{(k)}$ prostoru $\Lambda_{k,h}$, představuje $E_k \phi_j^{(k)}$ bázovou funkci prostoru $X_{k,h}$ jejichž restrikce na Γ_k je totožná s původní $\phi_j^{(k)}$.

Diskrétní slabá formulace pro Internodes úlohy (5a), (5b), (5c), (5d) je: Najít $u_{k,h} \in X_{k,h}$ splňující $u_{k,h} = g_{D,h}$ na Ω_D, k tak, že

$$a_k(u_{k,h}, v_{k,h}^0) = f_k(v_{k,h}^0), \quad \forall v_{k,h}^0 \in V_{k,h}^0, \quad k = 1, 2 \quad (45)$$

$$\Pi_{21} u_{1,h} - u_{2,h} = 0, \quad \text{na } \Gamma_2 \quad (46)$$

$$(r_u)_{1,h} + \Pi_{12}(r_u)_{2,h} = 0, \quad \text{na } \Gamma_1, \quad (47)$$

kde značením $u_{k,h}$ je myšleno $u_{h|\Omega_k}$, a $(r_u)_{k,h} \in \Lambda_{k,h}$ jsou diskrétní rezidua na rozhraní Γ_k , které jsou definovány

$$(r_u^{(k)})_i = a_k(u_{k,h}, E_k \phi_i^{(k)}) - f_k(E_k \phi_i^{(k)}), \quad i = 1, \dots, N_K, \quad (48)$$

$$(z_u^{(k)})_j = \sum_{i=1}^{N_k} (\mathbf{M}_{\Gamma_k}^{-1})_{ji} (r_u^{(k)})_i, \quad j = 1, \dots, N_K, \quad (49)$$

$$(r_u)_{k,h} = \sum_{j=1}^{N_k} (z_u^{(k)})_j \phi_j^{(k)}, \quad (50)$$

kde \mathbf{M}_{Γ_k} je matice hmotnosti na rozhraní Γ_k . Poznamenejme, že rovnice (47) je ekvivalentní [14] s podmínkou na rovnost derivací na rozhraní (5d). Pro více podrobností odkazujeme na [15].

4.4 Algebraická formulace

Pro výpočetní podoblasti definujeme sadu indexových množin

$$\begin{aligned} \mathcal{I}_{\bar{\Omega}_k} &= \{1, \dots, N_k\}, \\ \mathcal{I}_k &= \{i \in \mathcal{I}_{\bar{\Omega}_k} : x_i \in \bar{\Omega}_k \setminus (\partial\bar{\Omega}_{D,k} \cup \Gamma_k)\}, \\ \mathcal{I}_{D_k} &= \{i \in \mathcal{I}_{\bar{\Omega}_k} : x_i \in \partial\bar{\Omega}_{D,k}\}, \\ \mathcal{I}_{\bar{\Gamma}_k} &= \{i \in \mathcal{I}_{\bar{\Omega}_k} : x_i \in \bar{\Gamma}_k\}, \\ \mathcal{I}_{\Gamma_k} &= \{i \in \mathcal{I}_{\bar{\Omega}_k} : x_i \in \bar{\Gamma}_k \setminus \partial\bar{\Omega}_{D,k}\}, \\ \mathcal{I}_{\Gamma_k^D} &= \{i \in \mathcal{I}_{\bar{\Omega}_k} : x_i \in \bar{\Gamma}_k \cap \partial\bar{\Omega}_{D,k}\}. \end{aligned}$$

Je zřejmé, že pokud $\partial\Omega_D \cap \Gamma = \emptyset$, tak $\mathcal{I}_{\Gamma_k^D} = \mathcal{I}_{\Gamma_k}$. Mějme standardně definovanou matici tuhosti [18]

$$\mathbf{A}_{i,j}^{(k)} = a_k(\varphi_j^{(k)}, \varphi_i^{(k)}), \quad i, j \in \mathcal{I}_{\bar{\Omega}_k}. \quad (51)$$

Dále označme následující podmatice matice $\mathbf{A}^{(k)}$

$$\begin{aligned} \mathbf{A}_{kk} &= A^{(k)}(\mathcal{I}_k, \mathcal{I}_k), \\ \mathbf{A}_{k,\Gamma_k} &= A^{(k)}(\mathcal{I}_k, \mathcal{I}_{\Gamma_k}), \\ \mathbf{A}_{k,\bar{\Gamma}_k} &= A^{(k)}(\mathcal{I}_k, \mathcal{I}_{\bar{\Gamma}_k}), \\ \mathbf{A}_{\Gamma_k,k} &= A^{(k)}(\mathcal{I}_{\Gamma_k}, \mathcal{I}_k), \\ \mathbf{A}_{\bar{\Gamma}_k,k} &= A^{(k)}(\mathcal{I}_{\bar{\Gamma}_k}, \mathcal{I}_k), \\ \mathbf{A}_{\Gamma_k,\Gamma_k} &= A^{(k)}(\mathcal{I}_{\Gamma_k}, \mathcal{I}_{\Gamma_k}), \\ \mathbf{A}_{\bar{\Gamma}_k,\bar{\Gamma}_k} &= A^{(k)}(\mathcal{I}_{\bar{\Gamma}_k}, \mathcal{I}_{\bar{\Gamma}_k}). \end{aligned}$$

Podobně definujeme vektor neznámých a pravé strany

$$\mathbf{u}^{(k)} = [u_{k,h}(x_i^{(k)})], \quad \mathbf{f}^{(k)} = [f_k(\varphi_i^{(k)})], \quad i \in \mathcal{I}_{\overline{\Omega}_k} \quad (52)$$

a s nimi související „podvektory“, které vzniknou podobě jako podmatice $\mathbf{A}^{(k)}$

$$\begin{aligned} \mathbf{f}_k &= \mathbf{f}^{(k)}(\mathcal{I}_k), & \mathbf{f}_{\Gamma_k} &= \mathbf{f}^{(k)}(\mathcal{I}_{\Gamma_k}), & \mathbf{f}_{\overline{\Gamma}_k} &= \mathbf{f}^{(k)}(\mathcal{I}_{\overline{\Gamma}_k}), \\ \mathbf{u}_k &= \mathbf{u}^{(k)}(\mathcal{I}_k), & \mathbf{u}_{\Gamma_k} &= \mathbf{u}^{(k)}(\mathcal{I}_{\Gamma_k}), & \mathbf{u}_{\overline{\Gamma}_k} &= \mathbf{u}^{(k)}(\mathcal{I}_{\overline{\Gamma}_k}). \end{aligned}$$

Nakonec uvedeme „podvektory“ pro reziduum a funkci g_D

$$\begin{aligned} \mathbf{r}_{\Gamma_k} &= [(r_u^{(k)})_i], \quad i \in \mathcal{I}_{\Gamma_k}, & \mathbf{r}_{\overline{\Gamma}_k} &= [(r_u^{(k)})_i], \quad i \in \mathcal{I}_{\overline{\Gamma}_k}, \\ \mathbf{g}_{D_k} &= [g_D(\mathbf{x}_i^{(k)})], \quad i \in \mathcal{I}_{D_k}, & \mathbf{g}_{\Gamma_k^D} &= [g_D(\mathbf{x}_i^{(k)})], \quad i \in \mathcal{I}_{\Gamma_k^D}. \end{aligned}$$

Konečně se dostáváme k maticové formulaci Internodes. Pro porovnání nejdříve uvedeme soustavu pro konformní případ a potom ji zobecníme.

$$\begin{bmatrix} \mathbf{A}_{1,1} & \mathbf{O} & \mathbf{A}_{1,\Gamma_1} \\ \mathbf{O} & \mathbf{A}_{2,2} & \mathbf{A}_{2,\Gamma_2} \\ \mathbf{A}_{\Gamma_1,1} & \mathbf{A}_{\Gamma_2,2} & \mathbf{A}_{\Gamma_1,\Gamma_1} + \mathbf{A}_{\Gamma_2,\Gamma_2} \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_{\Gamma_1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 - \mathbf{A}_{1,D_1} \mathbf{g}_{D_1} \\ \mathbf{f}_2 - \mathbf{A}_{2,D_2} \mathbf{g}_{D_2} \\ \sum_{k=1,2} (\mathbf{f}_{\Gamma_k} - \mathbf{A}_{\Gamma_k,D_k} \mathbf{g}_{D_k}) \end{bmatrix} \quad (53)$$

Na rozdíl oproti metodě Mortarů, kde maticovou podobu sedlobodové úlohy „jen rozšíříme“ a neměníme zásadně její kompozici, zde matici soustavy (53) sestavujeme po podoblastech. Navíc se v této soustavě nevyskytuje stopa \mathbf{u}_{Γ_2} , protože je shodná se stopou \mathbf{u}_{Γ_1} . Z definice rezidua (48) můžeme poslední řádek (53) napsat jako $\mathbf{r}_{\Gamma_1} + \mathbf{r}_{\Gamma_2} = \mathbf{o}$. Tento řádek souvisí s podmínkou (5d).

Poznámka 5 Pro případ kdy platí $\partial\Omega \cap \Gamma \neq \emptyset$ musíme modifikovat definici rezidua (48) (viz. [15] rovnice (46)). Tato modifikace spočívá v přičtení matice $\mathbf{C}^{(k)}$, která je definovaná

$$\mathbf{C}_{ij}^{(k)} = - \int_{\partial\Omega_{D,k}} \partial_{L_k} \varphi_j^{(k)} \varphi_i^{(k)}, \quad i \in \mathcal{I}_{\Gamma_k^D}, \quad j \in \mathcal{I}_{\overline{\Omega}_k}, \quad (54)$$

kde ∂_{L_k} je normálová derivace a L je diferenciální operátor související s řešenou úlohou. V případě lineární eliptické úlohy (1) je $Lu = -\nabla \cdot (p(x)\nabla u) - q(x)u$ a $\partial_L u = p(x) \frac{\partial u}{\partial \mathbf{n}_\Omega}$. Modifikace rezidua se projeví tak, že $\mathbf{A}_{\overline{\Gamma}_k,G} = \mathbf{A}_{\overline{\Gamma}_k,G} + \mathbf{C}_{\overline{\Gamma}_k,G}^{(k)}$, kde $G \in \{D_k, \Gamma_k, \overline{\Gamma}_k, k\}$. Celá tato modifikace je popsána v [15].

Nyní přistoupíme k nekonformní maticové formulaci (45), (46), (47). Definujeme matice

$$\mathbf{Q}_{21} = \mathbf{R}_{21}, \quad \mathbf{Q}_{12} = \mathbf{M}_{\Gamma_1} \mathbf{R}_{12} \mathbf{M}_{\Gamma_2}^{-1}, \quad (55)$$

kde \mathbf{M}_{Γ_k} je matice hmotnosti na rozhraní Γ_k definovaná takto

$$(\mathbf{M}_{\Gamma_k})_{ij} = (\phi_i^{(k)}, \phi_j^{(k)})_2, \quad i, j = 1, \dots, N_k. \quad (56)$$

Matice \mathbf{Q}_{21} a \mathbf{Q}_{12} použijeme pro „komunikaci“ mezi uzly na rozhraní Γ . Algebraické podoba podmínky (46) je $\mathbf{Q}_{21}\mathbf{u}_{\bar{\Gamma}_2} - \mathbf{u}_{\bar{\Gamma}_1} = \mathbf{o}$. Podobně algebraické podoba podmínka na rovnost derivací (47) je $\mathbf{r}_{\bar{\Gamma}_1} + \mathbf{Q}_{12}\mathbf{r}_{\bar{\Gamma}_2} = \mathbf{o}$. Zavedené pojmy nyní použijeme v maticové formulaci Internodes pro nekonformní sítě

$$\begin{bmatrix} \mathbf{A}_{1,1} & \mathbf{O} & \mathbf{A}_{1,\Gamma_1} \\ \mathbf{O} & \mathbf{A}_{2,2} & \mathbf{A}_{2,\Gamma_2}\mathbf{Q}_{2,1}^0 \\ \mathbf{A}_{\Gamma_1,1} & \mathbf{Q}_{1,2}^0\mathbf{A}_{\Gamma_2,2} & \mathbf{A}_{\Gamma_1,\Gamma_1} + \mathbf{Q}_{12}^0\mathbf{A}_{\Gamma_2,\Gamma_2}\mathbf{Q}_{21}^0 \end{bmatrix} \begin{bmatrix} \mathbf{u}_1 \\ \mathbf{u}_2 \\ \mathbf{u}_{\Gamma_1} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \mathbf{f}_2 \\ \mathbf{f}_{\Gamma_1} + \mathbf{Q}_{12}^0\mathbf{f}_{\Gamma_2} \end{bmatrix} - \mathbf{G}, \quad (57)$$

kde

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \\ \mathbf{G}_{\Gamma_1} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{1,D_1}\mathbf{g}_{D_1} \\ \mathbf{A}_{2,D_2}\mathbf{g}_{D_2} \\ \mathbf{A}_{\Gamma_1,D_1}\mathbf{g}_{D_1} + \mathbf{Q}_{12}^0(\mathbf{A}_{\bar{\Gamma}_2,D_2}\mathbf{g}_{D_2} + \mathbf{A}_{\bar{\Gamma}_2,\Gamma_2}\mathbf{Q}_{21}^D\mathbf{g}_{D_1}), \end{bmatrix} \quad (58)$$

kde $\mathbf{Q}_{21}^0 = \mathbf{Q}_{21}(\mathcal{I}_{\bar{\Gamma}_2}, \mathcal{I}_{\Gamma_1})$, $\mathbf{Q}_{12}^0 = \mathbf{Q}_{12}(\mathcal{I}_{\Gamma_1}, \mathcal{I}_{\bar{\Gamma}_2})$ a $\mathbf{Q}_{21}^D = \mathbf{Q}_{21}(\mathcal{I}_{\Gamma_2}, \mathcal{I}_{\Gamma_1^D})$. Matice \mathbf{G} představuje začlenění nehomogenní Dirichletovy okrajové podmínky. V případě homogenní Dirichletovy okrajové podmínky je matice \mathbf{G} nulová. Poznamenejme, že pokud budeme uvažovat konformní úlohu, budou matice \mathbf{Q}_{21} a \mathbf{Q}_{12} rovny jednotkové matici. Z toho plyne, že soustava (57) přejde zpátky na soustavu (53).

4.5 Implementace

Protože jsme stěžejní pojmy již popsali v předchozích podkapitolách, uvedeme zde pouze algoritmus, který jsme naimplementovali v programovém prostředí MATLAB a jsou součástí přílohy této práce.

Procedura sestavující matice \mathbf{R}_{21} , \mathbf{R}_{12} je v příloze pod názvem *inter.m*. Procedura sestavení matice hmotnosti \mathbf{M} je v souboru *MKPM.m*. Sestavení matic \mathbf{Q}_{21} a \mathbf{Q}_{12} je implementováno vždy v konkrétním souboru s příkladem, kde je také implementován celý Algoritmus 3.

Algorithm 3 Internodes

- 1: sestav globální matici tuhosti \mathbf{A}
 - 2: **for** $k=1,2$ **do**
 - 3: urči indexové množiny $\mathcal{I}_k, \mathcal{I}_{\Gamma_k}, \mathcal{I}_{\bar{\Gamma}_k}, \mathcal{I}_{D_k}, \mathcal{I}_{\Gamma_k^D}$ a X
 - 4: sestav potřebné podmatice matice $\mathbf{A}^{(k)}$
 - 5: **if** $\partial\Omega_D \cap \Gamma \neq \emptyset$ **then**
 - 6: sestav matici $\mathbf{C}^{(k)}$ a modifikuj $\mathbf{A}_{\bar{\Gamma}_k, X}$
 - 7: **end if**
 - 8: sestav vektory $\mathbf{f}^{(k)}, \mathbf{g}_{D_k}$ a $\mathbf{g}_{\Gamma_k^D}$
 - 9: sestav matici hmotnosti \mathbf{M}_{Γ_k}
 - 10: **end for**
 - 11: sestav matice \mathbf{R}_{21} a \mathbf{R}_{12}
 - 12: sestav matice \mathbf{Q}_{21} a \mathbf{Q}_{12}
 - 13: sestav a vyřeš soustavu (57)
-

5 Numerické experimenty

V této části se zaměříme na předvedení funkcionality naší implementace. Uvedeme výsledky numerických experimentů obou výše popsaných metod. Začneme s experimenty na modelu membrány a poté ukážeme experimenty s lineární elasticitou [18].

5.1 Membrána

Zde uváděný model membrány je popsán Poissonovou rovnicí s homogenní okrajovou podmínkou

$$\begin{aligned} -\Delta u &= f \text{ na } \Omega, \\ u &= 0 \text{ na } \partial\Omega_D, \\ \frac{\partial u}{\partial n_\Omega} &= 0 \text{ na } \partial\Omega \setminus \partial\Omega_D, \end{aligned}$$

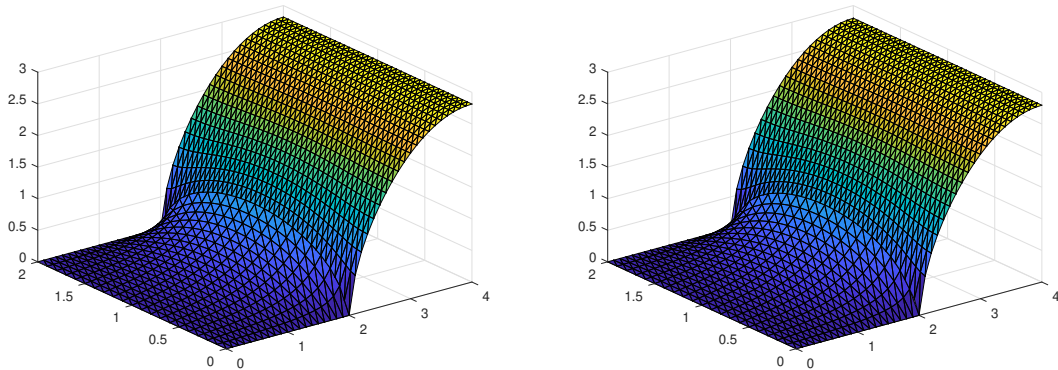
kde u je neznámá funkce a f je funkce zatížení. Ve všech experimentech jsou použity pro diskretizaci výpočetní oblasti lineární trojúhelníkové konečné prvky.

5.1.1 Experiment 1

Nechť $\Omega = (0, 4) \times (0, 4) \subset \mathbb{R}^2$ uvažujme dělení na $\Omega_1 = (0, 2) \times (0, 4)$ a $\Omega_2 = (2, 4) \times (0, 4)$. Dirichletova hranice je $\partial\Omega_D = \overline{\Omega}_1 \setminus \Omega$. Funkce zatížení je

$$f(x) = \begin{cases} 0, & x \in \Omega_1, \\ 1, & x \in \Omega_2. \end{cases}$$

V tomto experimentu jsou podoblasti Ω_1 a Ω_2 diskretizovány na 800 a 1600 trojúhelníků s 441 a 861 uzly. Hrany, které jsou lepeny mají 21 a 41 uzlů. Jak můžeme vidět na Obrázku 16, obě



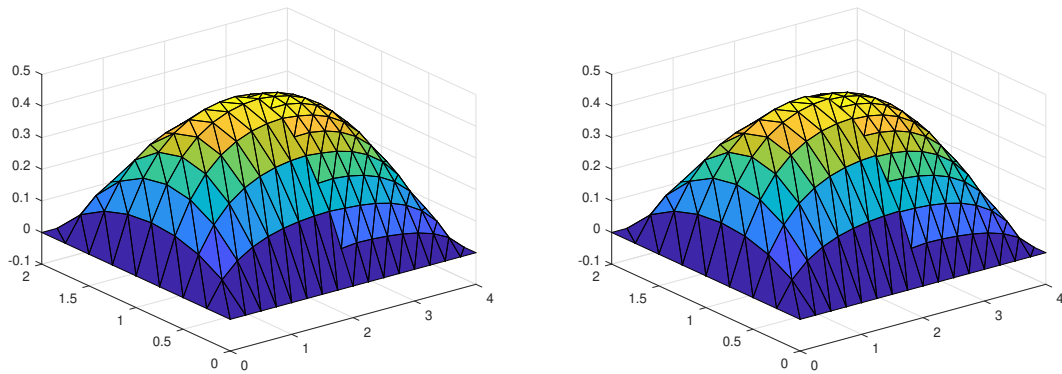
Obrázek 16: Experiment 1; Mortar (vlevo), Internodes (vpravo)

metody lepší podoblasti k sobě. Navíc na tomto experimentu můžeme pozorovat, že implemen-

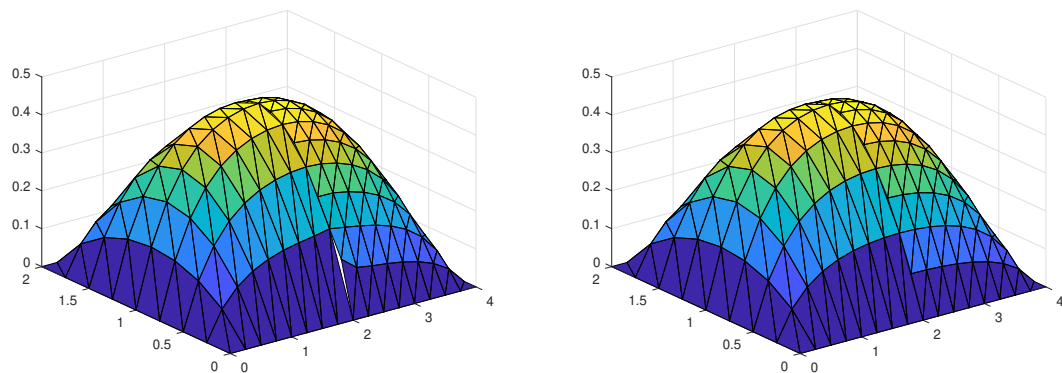
tované metody splňují podmínku spojitosti řešení i derivate řešení na hranici. Zvolili jsme tento experiment, abychom demonstrovali funkcionální obou přístupů. Protože, kdyby jedna z metod například nesplnila spojitost derivate, tak by řešení na hranici objevil „zlom“. V případě nesplnění rovnostní podmínky na rozhraní by výsledek byl nespojitý i v řešení. Tímto experimentem jsem ukázali, že obě metody jsou funkční. Výsledky jsou v obou případech prakticky shodné.

5.1.2 Experiment 2

Nechť $\Omega = (0, 4) \times (0, 4) \subset \mathbb{R}^2$ uvažujme dělení na $\Omega_1 = (0, 2) \times (0, 4)$ a $\Omega_2 = (2, 4) \times (0, 4)$. Dirichletova hranice je $\partial\Omega_D = \partial\Omega$. Funkce zatížení je $f = 1$. Tento experiment se zaměřuje na aproximaci na hranici podoblastí a rozdíly ve volbě master a slave hrany. V tomto experimentu jsou podoblasti Ω_1 a Ω_2 diskretizovány na 128 a 256 trojúhelníků s 81 a 153 uzly. Hraný, které jsou lepeny mají 9 a 17 uzlů.



Obrázek 17: Experiment 2: metoda Mortarů pro různé nastavení kombinace master-slave hran



Obrázek 18: Experiment 2: metoda Internodes pro různé nastavení kombinace master-slave hran

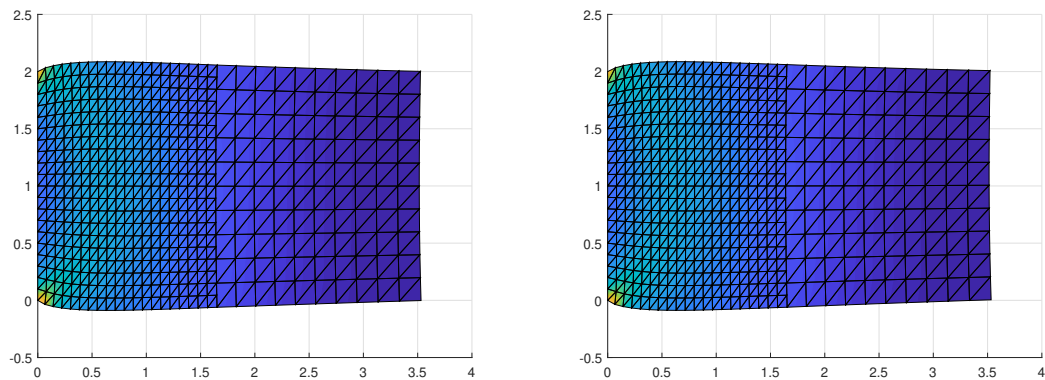
Na Obrázku 17 vlevo je master hrana na hrubší síti, můžeme vidět že, uzly ze slave hrany se přesně přizpůsobují master elementům. Na Obrázku 17 vpravo je master hrana na jemnější síti, můžeme vidět že, se uzly z obou stran snaží přizpůsobit pomyslenému průměru mezi sítěmi, nedochází k těsnému kopírování hodnot jako v případě vlevo.

Na Obrázku 18 vlevo je master hrana na hrubší síti, můžeme vidět že, uzly ze slave hrany nepřiléhají v blízkosti hranice rozhraní. Na Obrázku 18 vpravo je master hrana na jemnější síti, můžeme vidět že, se uzly z obou stran snaží přizpůsobit pomyslenému průměru mezi sítěmi, nedochází k těsnému kopírování hodnot jako v případě vlevo. Všechny experimenty však až na 18 vlevo splňují podmínky spojitosti na hranici.

5.2 Lineární elasticita

Implementaci modelu lineární elasticity jsem provedli podle knihy [18] (kapitola 9). Uvažovaný model předpokládá izotropní materiál popsany Lamého konstantami. Konkretní hodnoty pro všechny experimenty jsou dány $\lambda = \frac{\nu E}{(1-2\nu)(1+\nu)}$ a $\mu = \frac{E}{2(1+\nu)}$, kde $E = 2900$ a $\nu = 0.32$.

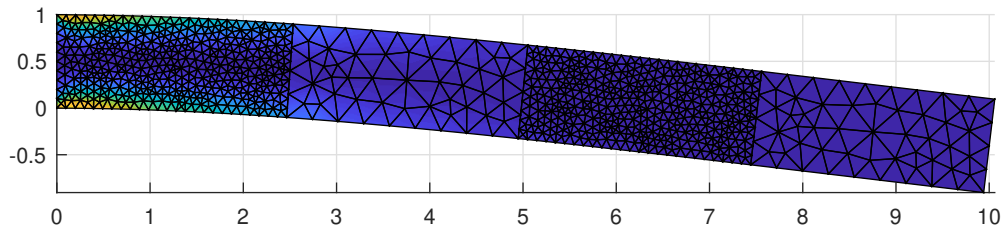
5.2.1 Experiment 3



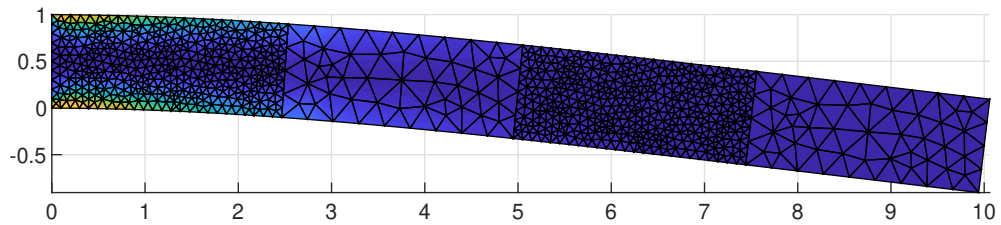
Obrázek 19: Experiment 3

Z Obrázku 19 můžeme vidět, že i v případě lineární elasticity jsou implementace obou metod funkční. V tomto experimentu je obdélníková oblast $\Omega = (0, 4) \times (0, 2)$ tlačena objemovou silou $f(x, y) = (-200, 0)$ proti hraně vlevo s homogenní Dirichletovou okrajovou podmínkou v obou směrech. Oblast je rozdělena na $\Omega_1 = (0, 2) \times (0, 2)$ a $\Omega_2 = (2, 4) \times (0, 2)$. V tomto experimentu jsou podoblasti Ω_1 a Ω_2 diskretizovány na 800 a 200 trojúhelníků s 441 a 121 uzly. Hrany, které jsou lepeny mají 21 a 11 uzlů. Dále můžeme v obou případech vidět očekávanou symetrii výsledků podle osy procházející bodem $[0, 1]$ rovnoběžné s vodorovnou souřadnicovou osou. V obou případech jsou splněny podmínky na rozhraní podoblastí.

5.2.2 Experiment 4



Obrázek 20: Experiment 4: metoda Mortarů



Obrázek 21: Experiment 4: metoda Internodes

Poslední experiment má simulovat ohyb tyčového objektu $\Omega = (0, 10) \times (0, 1)$, který je rozdělený na čtyři podoblasti, tak že se střídají dva parametry sítí $h_1 = h$, $h_2 = 3h$, kde $h = 0.1$. Oblast je rozdělena na $\Omega_1 = (0, 2.5) \times (0, 1)$, $\Omega_2 = (2.5, 5) \times (0, 1)$, $\Omega_3 = (5, 7.5) \times (0, 1)$ a $\Omega_4 = (7.5, 10) \times (0, 1)$. V tomto experimentu jsou podoblasti Ω_1 a Ω_3 diskretizovány na 810 trojúhelnících s 441 uzly. Podoblasti Ω_2 a Ω_4 jsou diskretizovány na 94 trojúhelnících s 61 uzly. Objekt je ukotven hranou vlevo. Působením objemové síly $f(x, y) = (0, -0.2)$ je objekt ohýbán směrem dolů. Na Obrázku 20 je ukázán výsledek metody Mortarů a na Obrázku 21 výsledek metody Internodes. Experiment znova potvrzuje funkcionalitu naší implementace. Oba výsledky jsou prakticky totožné a splňují podmínky na všech třech rozhraních.

6 Závěr

Téma této diplomové práce byla implementace metody Mortarů a Internodes. Obě metody byly teoreticky popsány a doplněny o praktické implementace v prostředí MATLAB. Kromě samotných metod byly implementovány procedury pro generování MKP soustav pro model membrány a lineární elasticity. Funkcionalita těchto metod lepení byla experimentálně otestována na zmíněných modelech.

Nyní shrneme poznatky získané studiem obou metod. Začneme metodou Mortarů. Metoda je z pohledu implementace rozsáhlejší, než metoda Internodes. Je třeba říct, že tato metoda spočívá v programové konstrukci projekce z jedné hrany rozhraní na druhou. Samotné řešení projekce a segmentace je programově trochu zdlouhavé, ale při dostatečném pochopení dané problematiky není v principu složité. Protože PERMON počítá se sedlobodovou formulací zpracovávaného problému, nebylo by začlenění metody Mortarů do softwareového balíčku po přepsání implementace z MATLABu složité. Výsledkem metody Mortarů jsou matice D a M , které se do soustavy přidají začleněním do vazební matice sedlobodové úlohy.

Metoda Internodes má z programátorského pohledu, kratší implementaci, která je však zároveň mnohem složitější. Metoda je založena na interpolaci z jedné hrany rozhraní na opačnou, ale také zpět. Oproti Mortarům Internodes používá dva operátory, pro vytvoření vazeb na rozhraní sítí. Slabina metody je podle mého názoru programově složitá implementace sestavování globální soustavy a použití mnoha různých indexací. Na druhou stranu část metody věnující se interpolaci není implementačně složitá. Z hlediska začlenění do softwareového balíku PERMON je metoda Internodes méně vhodná.

Dostupná literatura pro metodu Mortarů je mnohem rozsáhlejší, než pro metodu Internodes. Důvodem může být, že Internodes je poměrně nová metoda. Z uvedených důvodů je Mortar metoda pro implementaci do PERMONu vhodnější, než Internodes.

Kvůli současnému stavu ve světě jsme se dohodli s vedoucím práce na provedení implementace pro PERMON po odevzdání této práce. Navazující práce by poté mohla pokračovat paralelní implementací obou metod, popřípadě rozšířením pro 3D úlohy.

Literatura

- [1] Faker Ben Belgacem and Yvon Maday. The mortar element method for three dimensional finite elements. *Journal of Multivariate Analysis - MA*, 31, 01 1997.
- [2] Radim Blaheta. *Matematické modelování a metoda konečných prvků*. Vysoká škola báňská - Technická univerzita Ostrava, 2012.
- [3] Daniele Boffi, Franco Brezzi, and Michel Fortin. *Mixed Finite Element Methods and Applications*. Springer Berlin Heidelberg, 2013.
- [4] Martin D. Buhmann. *Radial Basis Functions: Theory and Implementations*. Cambridge Monographs on Applied and Computational Mathematics. Cambridge University Press, 2003.
- [5] A. T. Patera C. Bernardi, Y. Maday. A new nonconforming approach to domain decomposition: The mortar element method. in *Coll'ège de France Seminar, H. Brezis and J.-L. Lions, eds., Longman Scientific and Technical, Harlow, UK, 1994, pp. 13–51, 1994*.
- [6] Xiao-chuan Cai, Maksymilian Dryja, and Marcus Sarkis. Overlapping non-matching grid mortar element methods for elliptic problems. *SIAM Journal on Numerical Analysis*, 36, 05 1998.
- [7] Long Chen. Introduction to finite element methods, 2019, [online] [cit. 2020-03-23]. Dostupné z: <https://www.math.uci.edu/~chenlong/226/Ch2FEM.pdf>.
- [8] Simone Deparis, Davide Forti, Paola Gervasio, and Alfio Quarteroni. Internodes: an accurate interpolation-based method for coupling the galerkin solutions of pdes on subdomains featuring non-conforming interfaces. *Computers Fluids*, 141, 03 2016.
- [9] Zdeněk Dostal, David Horák, and Radek Kučera. Total FETI – an easier implementable variant of the feti method for numerical solution of elliptic pde. *Communications in Numerical Methods in Engineering*, 22:1155 – 1162, 12 2006.
- [10] Herbert Egger. A class of hybrid mortar finite element methods for interface problems with non-matching meshes, Technical Report AICES-2009-2, 2009.
- [11] Herbert Egger. Finite element methods for saddlepoint problems with application to darcy and stokes flow. In *The 11th Japanese-German International Workshop on Mathematical Fluid Dynamics*, 2015.
- [12] Charbel Farhat and Francois-Xavier Roux. A method of finite element tearing and interconnecting and its parallel solution algorithm. *International Journal for Numerical Methods in Engineering*, 32(6):1205–1227, 1991.

- [13] Yannis Fragakis and Manolis Papadrakakis. The mosaic of high performance domain decomposition methods for structural mechanics: Formulation, interrelation and numerical efficiency of primal and dual methods. *Computer Methods in Applied Mechanics and Engineering*, 192(35):3799 – 3830, 2003.
- [14] Paola Gervasio and Federico Marini. The internodes method for the treatment of non-conforming multipatch geometries in isogeometric analysis. *Computer Methods in Applied Mechanics and Engineering*, 358, 10 2019.
- [15] Paola Gervasio and Alfio Quarteroni. The internodes method for non-conforming discretizations of pdes. *Communications on Applied Mathematics and Computation*, 05 2019.
- [16] Václav Hapla, David Horák, Lukáš Pospíšil, Martin Čermák, Alena Vašatová, and Radim Sojka. Solving contact mechanics problems with permon. In *High Performance Computing in Science and Engineering*, pages 101–115, Cham, 2016. Springer International Publishing.
- [17] David Horák, Václav Hapla, Jakub Kružík, Radim Sojka, Martin Čermák, Jiří Tomčala, Marek Pecha, and Zdenek Dostal. A note on massively parallel implementation of feti for the solution of contact problems. *Advances in Electrical and Electronic Engineering*, 15, 07 2017.
- [18] David V Hutton. *Fundamentals of finite element analysis*. McGraw-hill, 2004.
- [19] Bouchala Jiří. *Variační metody*. Vysoká škola báňská - Technická univerzita Ostrava, 2012.
- [20] T. Kozubek, V. Vondrák, M. Menšík, D. Horák, Z. Dostál, V. Hapla, P. Kabelíkková, and M. Čermák. Total FETI domain decomposition method and its massively parallel implementation. *Advances in Engineering Software*, 60-61:14 – 22, 2013. CIVIL-COMP: Parallel, Distributed, Grid and Cloud Computing.
- [21] Dalibor Lukáš. Hierarchické konečné a hraniční prvky pro řešení maxwellových rovnic a tvarové optimalizace, 2010.
- [22] Seshaiyer Padmanabhan and Manil Suri. Hp submeshing via non-conforming finite element methods. *Computer Methods in Applied Mechanics and Engineering*, 189:1011–1030, 09 2000.
- [23] Zdeněk Dostál Petr Beremlijski. *Metody optimalizace*. Vysoká škola báňská - Technická univerzita Ostrava, 2012.
- [24] Dan Stefanica. Domain decomposition methods for mortar finite elements, 2000.
- [25] Dan Stefanica. A numerical study of feti algorithms for mortar finite element methods. *SIAM J. Scientific Computing*, 23:1135–1160, 2001.

- [26] Alena Vašatová, Jiří Tomčala, Radim Sojka, Marek Pecha, Jakub Kružík, David Horák, Václav Hapla, and Martin Čermák. Parallel strategies for solving the feti coarse problem in the permon toolbox. In *Programs and Algorithms of Numerical Mathematics*, pages 154–163, 06 2017.
- [27] Holger Wendland. Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree. *Adv. Comput. Math.*, 4(1):389–396, 1995.
- [28] Barbara Wohlmuth. A mortar finite element method using dual spaces for the lagrange multiplier. *SIAM Journal on Numerical Analysis*, 38, 01 2000.
- [29] Bin Yang, Tod Laursen, and Xiaonong Meng. Two dimensional mortar contact methods for large deformation frictional sliding. *International Journal for Numerical Methods in Fluids*, 62:1183–1225, 03 2005.